

Overføring av prosessdata via WAP



Diplomoppgave
for
sivilingeniørutdanning
innen
informasjons- og kommunikasjonsteknologi

av
Christer Bræck

Grimstad, mai 2000

Sammendrag

Denne oppgaven ser på mulighetene for å overføre prosessdata via en WAP-telefon. Prosessdataen som skal overføres er et EKG-signal. EKG er et signal som beskriver forandringer i elektrisk aktivitet på hjertet. En WAP-telefon skal i denne oppgaven fungere som en alarm som en hjertepasient hele tiden har med seg. Dersom pasienten slår alarm, skal WAP-telefonen hente ut pasientens EKG fra en database og sende disse til en alarmsentral, der det skal være mulig og følge pasientens utvikling.

Et EKG signal har en båndbredde fra 0-150 Hz. Det er mulig å kode dette signalet i forskjellige formater. Disse formatene har varierende kvalitet og i tillegg er båndbreddene forskjellige. Formatene som har blitt evaluert i denne oppgaven er WAV, MP3 og sampler. Når man skal overføre et EKG-signal må det også tas hensyn til tiden det tar å overføre et slikt signal. Det er viktig at ikke forsinkelsene i dataoverføringen blir for store.

Etter å ha vurdert de forskjellige formatene tror jeg det beste formatet for å overføre EKG-signaler er å bruke av sampler. Grunnen til dette valget er at overføring av EKG ved å bruke sampler gir den beste båndbredden. I teorien vil også kvaliteten i dette tilfellet bli bra. I tillegg gjør dette valget at det blir mulig å multiplekse flere signaler med tanke på båndbredden i GSM, som i dette tilfellet benyttes som bærer. Når det gjelder de andre formatene som ble evaluert, kommer det klart fram at WAV-filene blir for store og at MP3 filene er et grensetilfelle når man tar hensyn til tidskravene for overføring.

Siden WAP er en forholdsvis ny teknologi som kom på markedet i første kvartal år 2000, vil det i oppgaven også bli sett på WAP i framtiden. Nye versjoner av WAP-spesifikasjonen er klare, noe som gjør det mulig å se på mulighetene fremover. I tillegg vil det komme nye trådløse nettverk med mye større båndbredder en dagens GSM-nett. Det er ingen tvil om at WAP går en spennende fremtid i møte.

Forord

Oppgaven ”Overføring av prosessdata via WAP” ser på mulighetene for å overføre et EKG-signal via en WAP-telefon. WAP er en forholdsvis ny teknologi og det blir blant annet sett på om det er mulig å overføre prosessdata via WAP, med tanke på hvilke båndbredder som benyttes i WAP-teknologien.

Denne oppgaven er en del av et prosjekt der hensikten er å gjøre sykehuspasienter mest mulig trådløse. Den andre delen av prosjektet omhandler overføring av prosessdata via Bluetooth, som også blir omhandlet i en diplomoppgave ved Høgskolen i Agder.

Denne oppgaven er skrevet som et ledd i sivilingeniørutdanningen ved Høgskolen i Agder, avdeling for informasjons- og kommunikasjonsteknologi. Arbeidet med denne oppgaven har pågått i tidsrommet januar til og med mai år 2000. Jeg benytter også sjansen til å takke min veileder Rune Fensli for god og konstruktiv oppfølging gjennom hele prosjektet.

Grimstad, den 29. mai 2000

Christer Bræck

Innholdsfortegnelse

| | |
|---|-----------|
| Sammendrag..... | 2 |
| Forord | 3 |
| 1 Innledning..... | 6 |
| 1.1 Bakgrunn for oppgaven..... | 6 |
| 1.1.1 Opprinnelig oppgavebeskrivelse..... | 6 |
| 1.1.2 Utdypning av oppgavebeskrivelse | 6 |
| 1.1.3 Mål for oppgaven..... | 7 |
| 1.2 Dagens utbredelse av WAP | 8 |
| 1.3 Problemstillinger i oppgaven | 8 |
| 1.4 Litteratur | 9 |
| 1.5 Avgrensninger..... | 10 |
| 1.6 Rapportstruktur | 10 |
| 2 Bakgrunn for WAP..... | 12 |
| 2.1 Krav til WAP | 13 |
| 2.2 WAP-modellen | 13 |
| 2.3 Eksempel på WAP-nettverk..... | 14 |
| 3 Arkitekturen i WAP | 15 |
| 3.1 Applikasjonslaget (WAE)..... | 15 |
| 3.1.1 Agentene i WAE | 16 |
| 3.1.2 WTA | 16 |
| 3.2 Sesjonslaget (WSP)..... | 17 |
| 3.3 Transaksjonslaget (WTP)..... | 17 |
| 3.4 Sikkerhetslaget (WTLS) | 19 |
| 3.5 Transportlaget (WDP)..... | 19 |
| 3.5.1 Headerstruktur i WDP..... | 21 |
| 3.5.2 Segmentering og reassemblering | 22 |
| 3.5.3 WCMP (Wireless Control Message Protocol)..... | 23 |
| 4 WAP over GSM..... | 24 |
| 4.1 GSM..... | 24 |
| 4.1.1 Telefoni | 24 |
| 4.1.2 Kortmeldingstjenestene..... | 24 |
| 4.2 UDCP (USSD Dialogue Control Protocol)..... | 25 |
| 4.3 Adresseringen i WAP | 26 |
| 5 WAP i fremtiden | 28 |
| 5.1 GPRS..... | 28 |
| 5.2 UMTS | 29 |
| 5.3 De neste versjonene av WAP..... | 30 |

| | | |
|-----------|---|-----------|
| 6 | Beskrivelse av formater for overføring av data | 32 |
| 6.1 | WAV | 32 |
| 6.2 | MP3..... | 34 |
| 6.3 | Sampler | 36 |
| 7 | Metode..... | 37 |
| 7.1 | Systemskisse | 37 |
| 7.2 | Teoretisk løsningsforslag | 37 |
| 7.2.1 | EKG-signal | 37 |
| 7.2.2 | Databasene | 38 |
| 7.2.3 | Overføringsprosessen..... | 39 |
| 8 | Valg av verktøy | 42 |
| 8.1 | WAP utviklingsverktøy | 42 |
| 8.1.1 | WML..... | 42 |
| 8.2 | Databasene | 42 |
| 8.2.1 | ASP | 42 |
| 8.2.2 | SQL | 43 |
| 9 | Resultater..... | 44 |
| 10 | Drøfting og diskusjon | 48 |
| 10.1 | Drøfting av resultater | 48 |
| 10.1.1 | Databasene | 48 |
| 10.1.2 | Bruker grensesnittet | 48 |
| 10.1.3 | Overføringsformatet..... | 49 |
| 10.2 | Drøfting av verktøy..... | 49 |
| 10.3 | Forbedringer..... | 50 |
| 10.4 | Drøfting av WAP i fremtiden | 50 |
| 11 | Konklusjon..... | 52 |
| | Referanser..... | 53 |
| | Vedlegg A..... | 57 |
| | Conn.asp..... | 57 |
| | Vedlegg B | 58 |
| | Insert.asp | 58 |
| | Vedlegg C..... | 59 |
| | Query.asp | 59 |
| | Vedlegg D..... | 61 |
| | Forkortelser | 61 |

1 Innledning

1.1 Bakgrunn for oppgaven

1.1.1 Opprinnelig oppgavebeskrivelse

EKG (Electrocardiogram) er et apparat som måler forandringer i elektrisk aktivitet på hjertet. En WAP-telefon skal fungere som en alarm som en hjertepasient hele tiden har med seg. Dersom pasienten slår alarm skal WAP-telefonen hente ut pasientens EKG fra en database og sende disse til en alarmsentral, der det skal være mulig og følge pasientens utvikling.

Det skal undersøkes hvilke muligheter som finnes for å overføre analoge og digitale prosessdata via mobil kommunikasjon ved å benytte tjenester utviklet for WAP-løsninger.

Metoder for digitalisering og pakkesvitsjing skal evalueres med hensyn på mulig dynamisk båndbredde, og aktuelle komprimeringsteknikker skal beskrives og vurderes.

Det skal til slutt utvikles en testapplikasjon der man sender prosessdata via en WAP-telefon, og denne skal evalueres i forhold til de beskrevne teoretiske løsningsforslag.

1.1.2 Utdypning av oppgavebeskrivelse

Det vil bli tatt utgangspunkt i å benytte WAP-teknologien innen helsesektoren. Man antar at man har en pasient med hjerteproblemer som er i sitt eget hjem istedenfor på sykehus. Denne pasienten vil da ha en WAP-telefon i nærheten som vil fungere som en alarm. Dersom pasienten føler seg dårlig vil han/hun slå alarm via WAP-telefonen. WAP-telefonen vil da koble seg opp mot en PC som inneholder data om pasientens EKG. Et oscilloskop viser aktiviteten i form av pulser. Disse pulsene er en elektrisk representasjon målt ved hjelp av elektroder festet på kroppen. Et EKG-signal er et signal med en båndbredde fra ca 0-150Hz [1: Jenkins, 1999]. Ved hjelp av WAP-teknologien vil da pasientens EKG-diagram hentes ut fra PC'en og sendes videre til en alarmsentral på et sykehus.



Figur 1: EKG-signal [18: Paceart Associates, L.P, 1999]

Slik som oppgaven er definert er ment som utgangspunkt for undersøkelsene og prototypen som skal utvikles. Man ville for eksempel ikke la en pasient med alvorlige hjerteproblemer være alene i hjemmet uten tilsyn. Allikevel er hovedpoenget med oppgaven å få overført prosessdata ved bruk av WAP-teknologien. Grunnen til at EKG er valgt er at denne oppgaven kan være til interesse for blant annet helsesektoren. Noe av den tekniske utviklingen innenfor helsesektoren går på å gjøre pasientene mest mulig trådløse. Det vil si at de skal kunne gå seg en tur på sykehuset, uten å være avhengig av å måtte ha med seg store og tunge måleapparater.

1.1.3 Mål for oppgaven

Etter oppgaven ble valgt ble det satt opp noen målsetninger med oppgaven. Hovedmålene som ble satt opp var

- Se på mulighetene og begrensningene som finnes i WAP, spesielt med hensyn på båndbredder og overføringskapasitet
- Undersøke mulighetene for å overføre prosessdata via WAP
- Evaluere formater for overføring av prosessdata via WAP

1.2 Dagens utbredelse av WAP

WAP er en forholdsvis ny teknologi. De første WAP-telefonene har akkurat kommet ut på markedet. Versjonen som brukes på nåværende tidspunkt er WAP versjon 1.1. Det regnes med at antall brukere av mobile kommunikasjonsenheter vil overstige en 530 millioner i løpet av år 2001. Det regnes med at dette tallet vil dobles i løpet av 2004 [24: WAP-forum, 1999]. De fleste av disse brukerne er potensielle brukere av WAP-teknologien.

Slik WAP-telefonene fremstår i dag finnes det en del begrensninger både på batterikapasitet, nettverk og displayet på WAP-telefonen. Disse begrensningene er faktorer som hele tiden er under utvikling på grunn av den innflytelsen mobil kommunikasjon har fått på samfunnet. Utviklingen går mot en raskere og bedre kommunikasjon og forbrukerne krever stadig mer av produsentene for mobile enheter.

Når det gjelder bedrifter som satser på markedsføring og salg via WAP, så har dette begynt litt forsiktig. Det er sannsynligvis mange bedrifter som vil se an utviklingen før de satser på WAP-teknologien. Salg og bestillinger over WAP er i startfasen. Det er mange mangler, blant annet på sikkerheten, som gjør at mange foreløpig er avventende til dette. En mangel som er viktig for sikkerheten er ende-til-ende-kryptering. Dette er en viktig funksjon med tanke på handel via WAP. I de neste versjonene av WAP, 1.2 og 2.0, vil mer sikkerhet, deriblant ende-til-ende-kryptering være innført i WAP-teknologien [23: Aslaksen 2000].

Det kan med stor sikkerhet fastslås at WAP vil øke i omfang når de nyere versjonene blir lagt ut for salg. Grunnen til dette er at WAP-telefonene blir billigere og at de får flere funksjoner som ikke finnes på dagens mobiltelefoner. Dette kan være blant annet avanserte kalenderfunksjoner og tilgang til tjenester som for eksempel billettbestilling. I tillegg reklameres det mye for WAP-telefoner for tiden, noe som også kan være med å øke salget.

1.3 Problemstillinger i oppgaven

Hovedproblemet i oppgaven blir å bruke WAP-teknologien til å overføre prosessdata. Signalet som skal overføres er et EKG-signal. Dette er et signal med en båndbredde fra ca 0-150 Hz [1: Jenkins, 1999]. Et sentralt spørsmål er hvordan dette signalet skal overføres. Skal man bruke GSM USSD eller skal man bruke GSM SMS for overføring?

Det finnes mange måter å presentere et signal på. Hvordan skal EKG-signalet presenteres? Det beste ville være en slags "live" overføring av signalet. Sykehuset (alarmsentralen) vil da hele tiden kunne følge pasientens utvikling til han/hun får hjelp. Hvordan skal man varsle alarmsentralen om at noe har skjedd? Svaret er

Sannsynligvis en varslingslyd i forkant av filene. Hvordan skal denne lyden implementeres i varslingsystemet?

Et alternativ når det gjelder overføringen er å dele opp signalet i perioder på 5-10 sekunder. Når pasienten da slår alarm blir signalet overført del for del. Dersom dette alternativet skal benyttes må signalet lagres i en database/server. Signalet må da hentes ut ved hjelp av en WAP-telefon for deretter å videresendes til alarmsentralen.

Et problem som må løses er i hvilket format dataene skal sendes. Blant formatene det kan velges i mellom er WAV, MP3 og overføring av sampler. Når det gjelder digitalisering og komprimering kan det være mulig å kode WAV-filene i MP3-format. Hva slags verktøy må man bruke for å gjøre dette og hva må man gjøre for å pakke de komprimerte filene opp? Dersom man velger WAV-filer må man også bestemme Størrelsen på hver fil. Kanskje kan de være så små at man slipper å kode dem.

Når det gjelder sampler må det også velges hvor nøyaktig signalet skal presenteres i alarmdatabasen. Dette avhenger av samplingshastighet og antall bits som brukes til å representere hvert sampel.

Problemstillingene rundt metoder for digitalisering, komprimering og pakkesvitsjing må også vurderes. Det finnes en sammenheng i disse problemstillingene. Om pakkesvitsjing er mulig og eventuelt lønnsomt er blant annet avhengig av båndbredden som de valgte formatene for overføring krever.

I tillegg kommer noen problemstillinger rundt valg av utstyr. Det er viktig å finne verktøy som støtter hverandre. Når det gjelder emulatorer blir utgangspunktet testutstyr fra Nokia eller Ericsson.

1.4 Litteratur

Under arbeidet med diplomoppgaven har en stor del av arbeidet vært å samle inn informasjon. Dette kapitlet beskriver hvordan dette er gjort. Mye av informasjonen som er hentet inn vil til slutt, sammen med resultatet, danne grunnlaget for konklusjonen på denne oppgaven.

På forhånd hadde jeg ingen spesielle kunnskaper om WAP. Siden WAP er en forholdsvis ny teknologi er det foreløpig ikke skrevet så mye på området. Dersom man foretar et søk om WAP på internett vil man få opp mange avisartikler og mye reklame for WAP-telefoner. Dette er informasjon som ofte ikke holder mål som kilder i en diplomoppgave.

For å få den informasjonen som man trenger for å skrive en diplomoppgave kan man for eksempel gå inn på websidene til WAP-forum. Der ligger det mye informasjon om WAP og i tillegg ligger de nyeste WAP-spesifikasjonene der. Til denne oppgaven er det bare enkelte deler av spesifikasjonen som vil være til nytte. Siden man på forhånd ikke kunne vite hvilke deler det var, måtte hele spesifikasjonen lastes ned fra denne

websiden. Innsamling av informasjon fra WAP-forums websider tok mange dager. WAP-forums websider var utrolig trege og en nedlastningshastighet lavere enn 0,1 KB/s var ikke uvanlig. Dette tar da meget lang tid når enkelte av dokumentene som skal lastes ned har en filstørrelse på opp til 4 MB. Det viste seg at enkelte dager og tider på døgnet var bedre for nedlasting av informasjon enn andre.

Siden WAP-teknologien kan benytte seg av flere bærertjenester var også en litteraturstudie i GSM-systemet nødvendig, siden GSM skulle benyttes som bærer i dette tilfellet. Denne teorien inngår som en del av faget Signalering i Telekommunikasjonssystemer.

1.5 Avgrensninger

For at oppgaven ikke skal bli alt for omfattende må man sette opp avgrensninger. I oppgaven er det tatt utgangspunkt i at det er en og bare en pasient. Denne personen befinner seg i utgangspunktet i sitt eget hjem. Siden denne pasienten befinner seg i sitt eget hjem så fjerner man behovet for roaming på GSM-nettet siden pasienten vil være knyttet opp mot en og samme basestasjon hele tiden.

Hvert av lagene i WAP-protokollen er beskrevet. Dette er ment som et grunnlag for å få et inntrykk av hvordan samhandlingen mellom de forskjellige lagene forgår. Det er i utgangspunktet transportlaget (WDP) som er viktig for denne oppgaven. Det som er skrevet om de andre lagene er derfor ikke veldig detaljert. Når det gjelder transportlaget er det som er viktig for å løse oppgaven beskrevet.

Denne rapporten baseres mest på teknologien WAP og ikke det samfunnsfaglige rundt oppgaven. For en sykehuspasient ville personvern vært meget viktig ved overføring av en pasients EKG. Dette er ikke tatt med i denne oppgaven. Grunnen til dette er at prosessdata kan være mye mer enn et EKG-signal og at prosessdataen ikke nødvendigvis bør ha noen tilknytning til sykehus eller pasienter.

Rapporten beskriver heller ikke mye om sikkerheten i WAP. Det som står om sikkerhet, står beskrevet i kapittel 3.4. Dette er et område som ikke er dekket da det er så omfattende at det ikke gir noen sammenheng dersom man trekker ut essensene. Det å dekke sikkerheten er en stor oppgave i seg selv, og jeg valgte å konsentrere meg om overføringsprosessen. Sikkerhetsproblematikken i WAP blir gjennomgått i diplomoppgaven "Security in WAP" [30: Storfjord, 2000].

1.6 Rapportstruktur

Rapporten begynner med en innledning. I innledningen kan man se hva oppgaven egentlig går ut på. Her vil problemstillinger rundt oppgaven bli tatt opp slik at leser får et inntrykk av hva som skal til for å løse oppgaven og eventuelt gjøre opp sine egne meninger om hvordan oppgaven bør løses.

Kapittel 2 heter "Bakgrunn for WAP" og her blir noe av WAP-teknologien gjennomgått uten detaljer. Hensikten med dette kapitlet er å gi leser en kort innføring i hvorfor WAP-teknologien er som den er. Her kan man se hva målene og motivene bak WAP er. Her blir også noen av de faktorene som begrenser WAP nevnt.

I kapittel 3 blir WAP gjennomgått i en mer detaljert form. Det som finnes her er utelukkende teori som er omarbeidet ut fra WAP-spesifikasjonen. Denne teorien er omarbeidet ut fra hvilke deler som må være med for å kunne løse oppgaven. Referansene i denne delen står som oftest i overskriftene. Grunnen til dette er at det som oftest bare er den referansen som er brukt for å omarbeide det aktuelle kapitlet (eventuelt underkapitlet). I referanselisten bak i rapporten er alt angitt så presist som mulig.

Kapittel 4 beskriver GSM som i denne oppgaven er brukt som bærer. Det er mye fakta om GSM-systemet i dette kapitlet og det har blitt lagt mest vekt på overføringskapasitet og båndbredder. I tillegg er det beskrevet litt om adresseringen i WAP.

I fremtiden vil WAP kunne benytte seg av bedre bærere. Raske nettverk som GPRS og UMTS er under utvikling og utprøving. Disse nettverkene blir beskrevet i kapittel 5. Samtidig vil man få en kort beskrivelse av hva som kan ventes av nye funksjonaliteter i de kommende versjoner av WAP.

En viktig del av oppgaven er å evaluere forskjellige kodeformater som prosessdataen som skal overføres kan kodes i. Disse kodeformatene blir presentert i kapittel 6.

Metodekapitlet er det teoretiske løsningsforslaget for oppgaven. Her trekkes viktige elementer som tidligere er presentert i oppgaven ut, for deretter å bli satt sammen til en applikasjon. I dette kapitlet blir teorier i de tidligere kapitlene i rapporten gjenspeilet ved å referere til nummerene på de forskjellige kapitlene.

Verktøyene som er brukt for å løse oppgaven blir beskrevet i kapittel 8. I kapittel 9 blir resultatene presentert.

I kapittel 10 blir oppgaven drøftet. Drøftingen danner mye av grunnlaget for konklusjon som kommer i det påfølgende kapitlet. Her blir resultatene drøftet med hensyn på metodedelen. Her kan man blant annet se om oppgaven ble løst slik den skulle, hvilke forandringer som ble gjort i forhold til metodedelen og hvilke problemer som dukket opp under applikasjonsutviklingen. Deretter blir verktøyene som ble brukt diskutert, før det blir sett nærmere på forbedringer og fremtidsperspektivet for WAP.

Til slutt har rapporten en liste med referanser som etterfølges av noen vedlegg.

2 Bakgrunn for WAP [19: WAP-forum, 1999]

WAP er et resultat av WAP-forums forsøk på spesifisere en internasjonal standard som vil være nyttig til utvikling av applikasjoner og tjenester som kan brukes over trådløse nettverk. WAP spesifiserer nettverksprotokoller og et rammeverk for applikasjoner til trådløse komponenter som for eksempel mobiltelefoner og PDA (Personal Digital Assistant).

Motivet bak WAP er:

- Få internett og andre avanserte datatjenester på små trådløse terminaler.
- Lage en global trådløs spesifikasjon som vil fungere på forskjellige trådløse nettverk.
- Lage applikasjoner som skalerer og som skal kunne ta i bruk flere typer bærere.
- Utnytte de standarder som eksisterer og videreutvikle disse der dette er nødvendig.

To områder der man kan se en rask utvikling er trådløs datakommunikasjon og internett. Med et WAP-interface vil man få disse områdene integrert i en trådløs enhet. Den økende bruken av trådløs kommunikasjon og internett gjør at det hele tiden er behov for eksisterende og nye informasjonstjenester.

Det meste av dagens internett teknologi er laget for vanlige PC'er med høy båndbredde, stor prosesseringskraft og nettverk som man kan stole på. Mobilenheter har helt andre forutsetninger en vanlige PC'er. Noe som vanligvis skiller mobilenheter fra vanlige PC'er er at de har:

- Mindre prosesseringskraft.
- Mindre minne (ROM og RAM)
- Begrenset strømforbruk
- Mindre display
- Færre tilkoblingsmuligheter.

Med WAP må det også tas hensyn til nettverkene som brukes. All data må gå over trådløse nettverk, noe som gjør det vanskeligere med tanke på overføring av store datamengder. På grunn av begrenset strømtilgang, frekvensområde og mobilitet har trådløse nettverk i forhold til faste nettverk vanligvis:

- Mindre båndbredde
- Større tidsforsinkelse
- Mindre stabilitet i forbindelsen
- Mindre forutsigbar tilgjengelighet

Mange av de nåværende nettverkene som er laget for mobil kommunikasjon tilbyr tilleggstjenester for å forbedre tjenestekvaliteten ytterligere. Disse nettverkene har likevel langt igjen i forhold til de faste nettverkene, spesielt når det gjelder båndbredder og tidsforsinkelser.

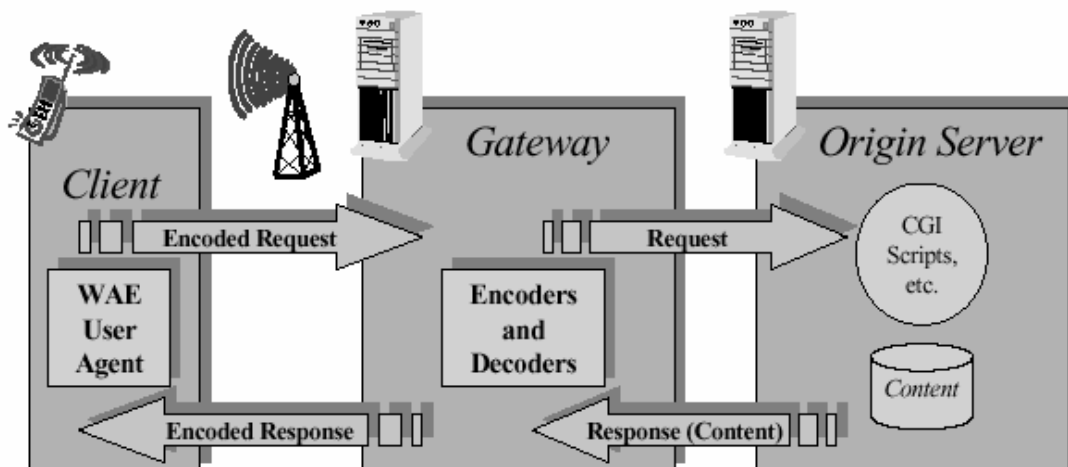
2.1 Krav til WAP

Da WAP-standarden ble utarbeidet var det mange krav som måtte tilfredsstilles. Et av kravene var at WAP skulle kunne bruke mange av de standardene som allerede eksisterte. Det fantes fra før mange forskjellige protokoller som var beregnet for overføring da første del av WAP-standarden ble utarbeidet. WAP skulle bygge på en lagdelt arkitektur som var skalerbar. Det er viktig å tenke på fremtidige teknologier når det utarbeides en standard. Med tanke på dette ble det bestemt at WAP skulle støtte så mange trådløse nettverk som mulig.

For at WAP skulle være brukervennlig måtte det blant annet lages applikasjoner som ikke tok så mye minne og prosesseringskraft. Resultatet var nye språk beregnet for mobilkommunikasjon. Disse språkene var mer effektive enn de språkene som brukes i internettssammenheng. Språkene var WML og WMLScript. Disse språkene ligner på HTML og JAVA [28: Telenor].

2.2 WAP-modellen

WAP-arkitekturen gjør at de som skal utvikle applikasjoner får en del fordeler. Som man kan se på figur 2 skiller ikke programmeringsmodellen for WAP seg så mye fra WWW-modellen. Arkitekturen er kjent og det er mulig å bruke kjent verktøy som for eksempel Web servere og XML. Optimalisering av verktøyene er viktig siden det handler om trådløs kommunikasjon. Der det har vært mulig har det som utgangspunkt blitt brukt standarder som ble laget før WAP-standarden. WAP-spesifikasjonene er spesifisert og basert på grunnlag av WWW-formatene. Data blir transportert ved bruk av standard kommunikasjonsprotokoller basert på kommunikasjonsprotokollene til WWW. En nettleser i den trådløse terminalen har ansvaret for å koordinere brukers interface, slik at bruker får et skjermbilde som er så enkelt som mulig å lese.

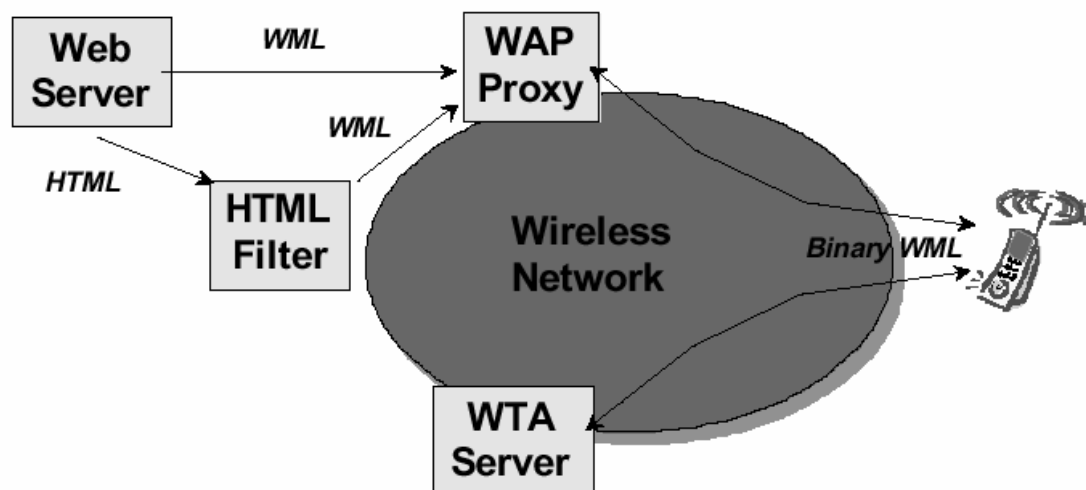


Figur 2: WAP programmeringsmodell [19: WAP-forum, 1999]

WAP har definert noen standard komponenter for å få til kommunikasjonen mellom mobilterminalene og nettverksserverne. Også i dette tilfellet ligger teknologien fra internett til grunn for tankegangen. For å identifisere bestemte lokasjoner og servere brukes standard internett navngiving. Siden WAP-standardene ofte er basert på teknologien fra internett, er det også laget standarder for kalenderinformasjon, kredittkortobjekter, bilder og programmeringsspråk i WAP.

WAP benytter seg av proxy-teknologi for å opprette forbindelsen mellom det trådløse nettverket og internett. En proxy som skal brukes av WAP må ha en protokoll gateway som oversetter forespørsler fra WAP sin protokollstack (WSP, WTP, WTLS og WDP) til protokollstacken for internett (HTTP og TCP/IP). En proxy må også inneholde komponenter for koding og dekoding av data for å få dataen som skal til WAP-terminalen så kompakt som mulig.

2.3 Eksempel på WAP-nettverk



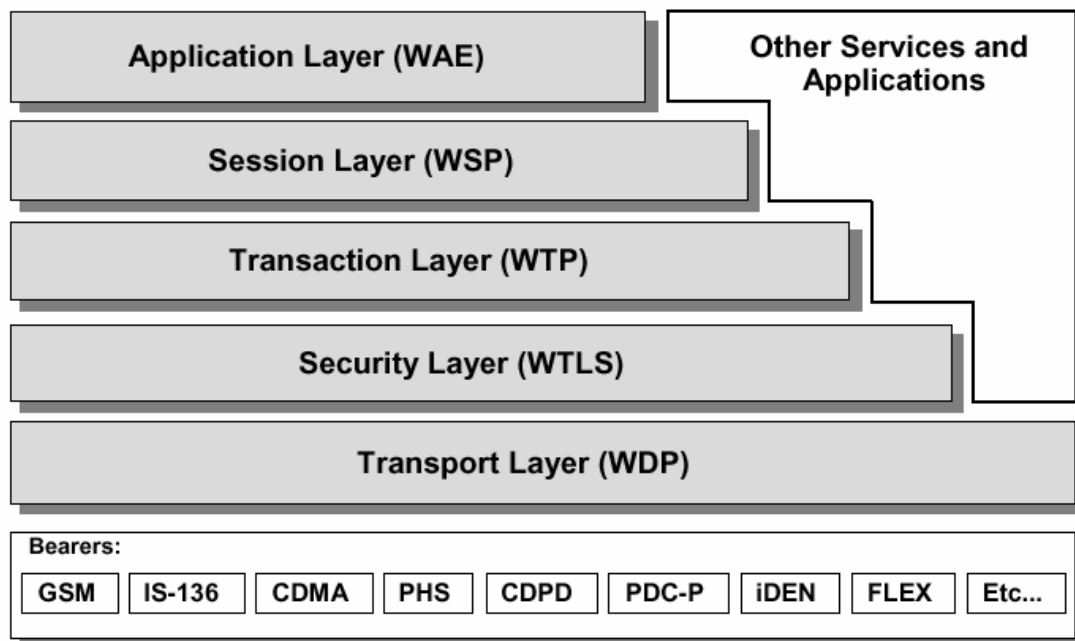
Figur 3: Eksempel på WAP-nettverk [19: WAP-forum, 1999]

I eksempelet på figur 3 kommuniserer WAP-klienten med to servere i det trådløse nettverket. Proxyen oversetter forespørsler i WAP-formatet (WML) til forespørsler i WWW-formatet (HTML). Proxyen vil deretter kode svarene på forespørslene tilbake i WAP-formatet som er binært.

Dersom webserveren skulle svare i WAP-formatet vil disse svarene bli sendt direkte til WAP-terminalen via proxyen. Skulle svarene derimot komme i WWW-formatet vil disse gå gjennom et HTML-filter der de blir konvertert fra HTML til WML.

WTA-serveren vil svare på forespørsler fra WAP-klienten umiddelbart etter at de er mottatt. Denne serveren sørger for at WAP har aksess mot for eksempel mobiltelefonnettet.

3 Arkitekturen i WAP



Figur 4: Protokollarkitekturen i WAP [19: WAP-forum, 1999]

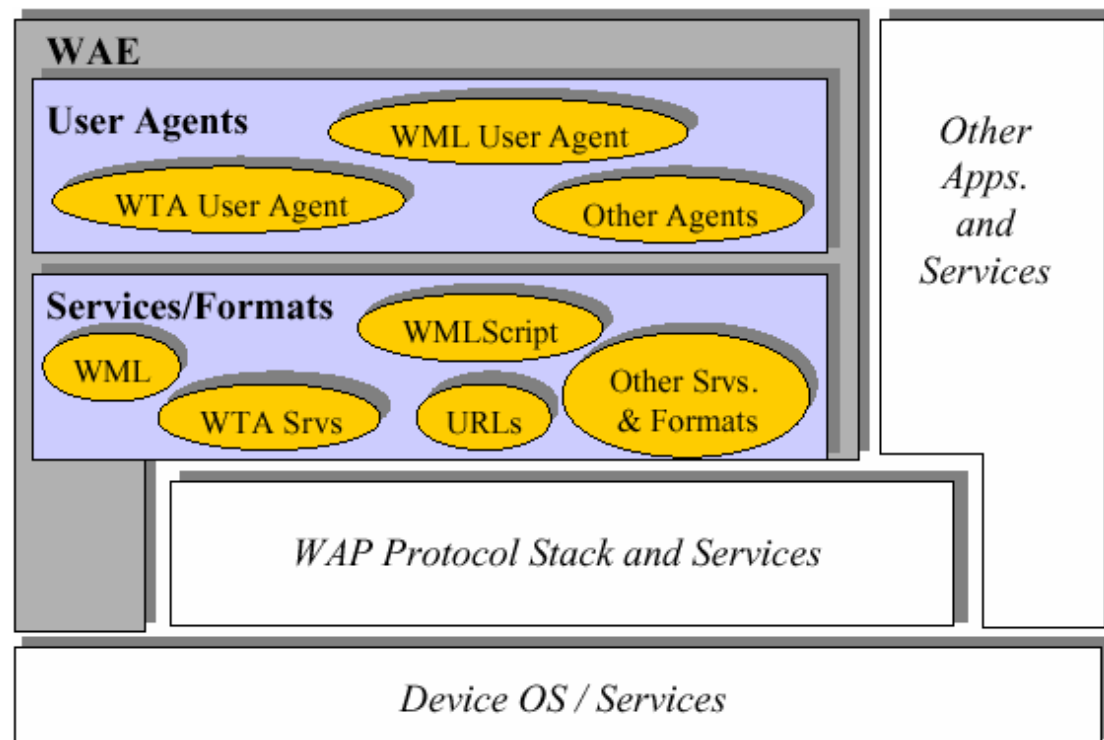
Arkitekturen i WAP er lagdelt og består som man kan se i figur 4 av 5 forskjellige lag. I tillegg kommer et lag med forskjellige bærere [19: WAP-forum, 1999]. Hvert av lagene kan aksesseres av laget som ligger over. Tjenester og applikasjoner kan aksessere de forskjellige lagene direkte ved hjelp av forskjellige interface.

3.1 Applikasjonslaget (WAE) [6: WAP-forum, 1999]

WAE spesifiserer et rammeverk for applikasjoner beregnet for trådløse komponenter som for eksempel mobiltelefoner og PDA. WAE bygger på teknologien og tankegangen fra internett. Målet med WAE er at det skal kunne lages tjenester og applikasjoner som kan operere på flere forskjellige trådløse plattformer på en effektiv måte.

WAE er designet på en slik måte at det støtter utvikling av interaktive applikasjoner som skal brukes på komponenter med begrenset kapasitet, både når det gjelder batterilevetid, skjermstørrelse og begrenset antall innganger. Det er også tenkt på at det skal være lettere å utvikle interaktive applikasjoner med tanke på større tidsforsinkelser enn det som er vanlig i fastnettet. Noe av det viktigste med WAE er nettopp at det støtter tidsforsinkelser og liten båndbredde.

WAE er som man kan se på figur 5 i utgangspunktet delt inn i to logiske lag. Det ene laget er brukeragentene. Dette laget inneholder blant annet nettlesere og telefonbøker. Det andre laget er en plattform for tjenester og formater som støttes i WAP. Dette inneholder elementer som er viktig for de som utvikler applikasjonene for brukerne. Disse applikasjonene kan aksesseres via brukeragentene og kan for eksempel være WML, WMLScript, bildeformater og kalenderformater.



Figur 5: Komponenter i WAE-arkitekturen [6: WAP-forum, 1999]

3.1.1 Agentene i WAE

En viktig agent i WAE er WML-nettleseren. En annen agent som står sentralt i WAE er WTA-agenten. Denne agenten har støttet mot mobiltelefonapplikasjoner som for eksempel telefonbok og kalenderfunksjoner. En fordel med WAE er at også agenter fra andre arkitekturer og miljøer kan integreres i denne protokollen.

WAE har ikke i seg selv spesifisert noen agenter, men grunnlaget for agentene er definert i WAE. Hvordan agentene skal være og fungere er opp til de som lager og produserer dem.

3.1.2 WTA

WTA er en av tjenestene i WAP som støtter mobiltelefoni. WTA er beregnet som et tillegg for de som ønsker flere tjenester. WTA integrerer datanettverkene og

nettverkene beregnet for lyd. Rammeverket i WTA støtter også sanntidsapplikasjoner til brukeren.

3.2 Sesjonslaget (WSP) [7: WAP-forum, 1999]

WSP er laget for å støtte blant annet tjenester for transaksjoner og datagram. Sikkerheten rundt disse tjenestene kan velges om nødvendig fra et eget lag, WTLS, som befinner seg ovenfor transportlaget. I mange tilfeller må det ekstra komponenter til for å få en sikker forbindelse. Denne sikkerheten vil ikke WSP protokollen automatisk sørge for. Grunnen til det er at WSP i seg selv ikke trenger noe sikkerhetslag. Sikkerheten på dette laget fungerer derfor slik at applikasjonene som krever ekstra sikre forbindelser må selv si i fra om dette.

En av WSP-protokollens oppgaver er å sørge for organisert utveksling av data mellom klient/server-applikasjoner. WSP skal opprette en sikker sesjon mellom klient og server, og deretter holde orden på sesjonene. En viktig oppgave med tanke på WAP-terminalenes forutsetninger er kodingen. WSP skal sørge for utveksling av data mellom klient og server med en kodeform som er kompakt nok til å kunne håndteres av WAP-terminalene. Noen ganger må sesjoner og transaksjoner avbrytes eller utsettes. Det er WSP-protokollens oppgave å avslutte sesjoner på en slik måte at de kan gjenopptas uten at for mye data som må overføres på nytt.

Designen i WSP er HTTP på binær form. Dette gjør at forespørsler til serveren og svar til klienten kan inneholde både headere og annen data. For at WSP skal være fullt ut kompatibel med HTTP/1.1 er det i tillegg mulig å forhandle om typer av forespørsler. WSP tar seg av dataoverføringer for applikasjonslaget. Headerne i HTTP/1.1 brukes blant annet til å definere type, kodeform og språk. For å redusere bruken av metadata finnes det mer kompakte former for headere på binær form. WSP spesifiserer også et kompakt dataformat som minner om MIME "multipart/mixed" som brukes av HTTP/1.1.

Det er ikke WSP selv som legger headere på forespørslene og tilbakemeldingene. WSP vil passere klientens og serverens sesjons-, request- og responseheadere uten at noen ekstra headere blir lagt til eller fjernet.

3.3 Transaksjonslaget (WTP) [8: WAP-forum, 1999]

Det finnes tre typer transaksjonstjenester i WTP. Disse er definert i klasser.

- Klasse 0: Upålitelig anrop uten svar
- Klasse 1: Pålitelig anrop uten svar
- Klasse 2: Pålitelig anrop med eksakt ett pålitelig svar

Pålitelighet i overføringene oppnåes blant annet ved bruk av unike identifikatorer under transaksjoner, tilbakemeldinger og retransmisjoner. I WTP finnes ingen faser for å sette opp eller ta ned forbindelser. Disse fasene utelates for å slippe unødige mye header informasjon på forbindelsen.

Det er også mulig å velge ekstra tjenester i WTP. Bruker-til-bruker-pålitelighet kan velges og WTP vil da bekrefte hver eneste melding som mottas. Dersom man ønsker å se informasjon som er relatert til overføringen er også dette mulig. Dette kan for eksempel være overføringshastigheten som i enkelte tilfeller kan være viktig.

Figur 6 viser hvordan WTP-protokollen relateres til andre WAP-protokoller. En viktig del av WTP er at den har en mekanisme som gjør nummeret på antall transaksjoner som blir sendt på nytt så lite som mulig. Dette kan gjøres fordi mange transaksjoner som sendes flere ganger skyldes dupliserte datapakker, noe WTP har kontroll over. WTP-protokollen støtter i tillegg asynkrone transaksjoner og mottaker sender tilbake resultatet av forespørslene så fort de blir tilgjengelige.

| WTP User (e.g. WSP) | |
|---|--|
| WTP | <input type="checkbox"/> Transaction handling <input type="checkbox"/> Re-transmissions, duplicate removal, acknowledgements <input type="checkbox"/> Concatenation and separation |
| [WTLS] | <input type="checkbox"/> Optionally compression <input type="checkbox"/> Optionally encryption <input type="checkbox"/> Optionally authentication |
| Datagram Transport (e.g. WDP) | <input type="checkbox"/> Port number addressing <input type="checkbox"/> Segmentation and re-assembly (if provided) <input type="checkbox"/> Error detection (if provided) |
| Bearer Network (e.g. IP, GSM SMS/USSD, IS-136 GUTS) | <input type="checkbox"/> Routing <input type="checkbox"/> Device addressing (IP address, MSISDN) <input type="checkbox"/> Segmentation and re-assembly (if provided) <input type="checkbox"/> Error detection (if provided) |

Figur 6: WTP-protokollens relasjon til andre WAP-protokoller [8: WAP-forum, 1999]

Samarbeid med andre protokoller i WAP er viktig for WTP. WTP er spesifisert for å kunne benytte seg av bærertjenester beregnet for datagram¹. Datadelen fra WTP-protokollen kan finnes igjen i datadelen i datagrammet. Siden datagram er ustabile må WTP kunne retransmitere og sende bekreftelser på at datapakkene er mottatt for å kunne gi brukeren en pålitelig overføringstjeneste. WTP er også ansvarlig for å sette sammen data fra flere forskjellige protokoller og samle disse i en og samme dataenhet, dersom dette er mulig.

Det er bærerne som har ansvaret for å rute datagrammene til riktig destinasjon. Adresseringen er forskjellig i de forskjellige bærerne noe som kan være et problem. I tillegg bruker noen nettverk dynamisk adressering. Løsningen er å gå via en server og

¹ Et datagram er en datapakke som er forbindelsesløs. Med forbindelsesløs menes at datagrammet kan da forskjellige veier i nettet selv om destinasjonen er den samme.

spesifisere bærertypen før dataen blir sendt. Bærertjenesten kan som oftest bestemmes ved å se adressetypen.

Det finnes et interface mellom WTP og miljøet rundt WTP kalt WTP Mangement Entity. WTP Mangement Entity sørger for at WTP-protkollen får vite om forandringer som skjer i miljøet rundt WTP. Forandringer kan ha innflytelse på hvilken operasjon som WTP skal utføre.

3.4 Sikkerhetslaget (WTLS) [9: WAP-forum, 1999, 19: WAP-forum, 1999]

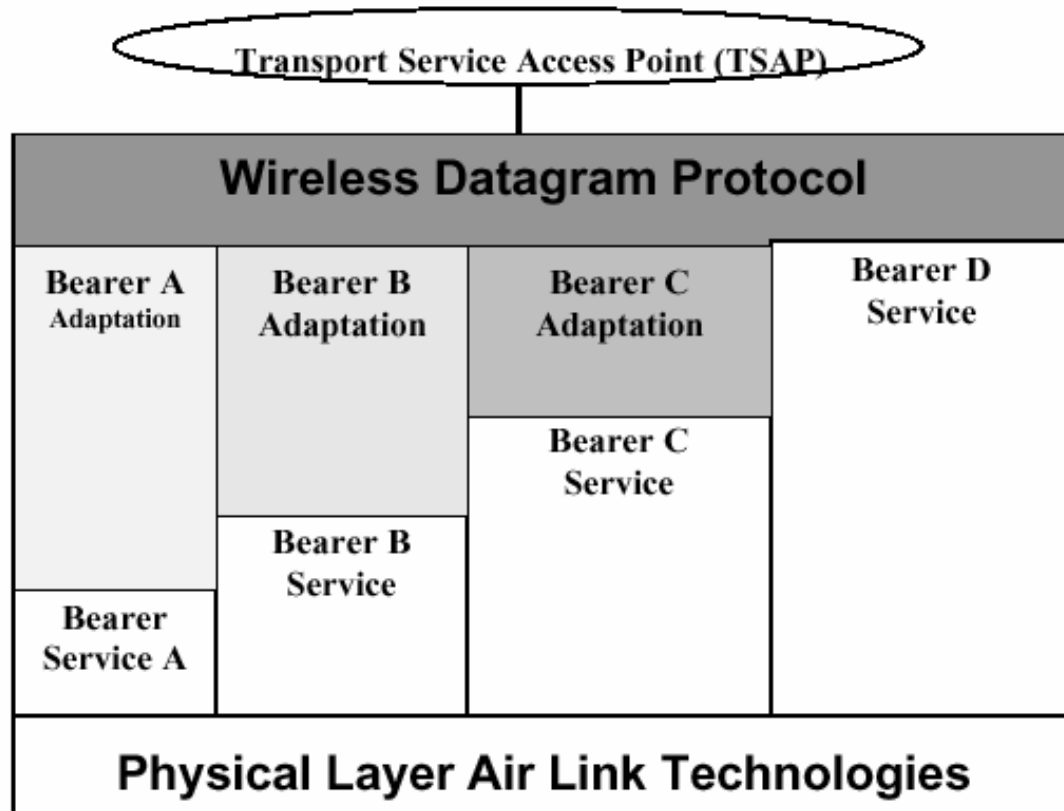
WTLS er en sikkerhetsprotokoll basert på industristandarden Transport Layer security (TLS). WTLS skal brukes sammen transportprotokollene i WAP og er derfor optimalisert med tanke på lav båndbredde og trådløs kommunikasjon.

En viktig oppgave for WTLS er dataintegritet. Det vil si at dataen som sendes i mellom terminalen og en applikasjonsserver må forbli uforandret. En ting som står sentralt i WTLS er kryptering av data. Dersom sender og mottaker ønsker det, skal WTLS sørge for at ingen klarer å tolke datastrømmen som blir sent, selv om de kan se dataene. I tillegg er også autentisering viktig. Det må være sikkert at dataene når den personen som skal ha de og ingen uvedkommende. Denne autentiseringen foregår også mellom terminalen og applikasjonsserveren.

En tjeneste som også støttes av WTLS er Denial-of-service. Dette vil si at WTLS kontrollerer innkommende forespørsler om å få tilgang til diverse applikasjoner. Dersom innkommende forespørsel ikke godkjennes sendes en melding i retur som sier at forespørselen ikke er godkjent. Den som ville ha tilgang til applikasjonen vil da ikke få dette. Denne typen tjeneste beskytter de øvre lagene mot angrep fra uvedkommende.

3.5 Transportlaget (WDP) [10: WAP-forum, 1999]

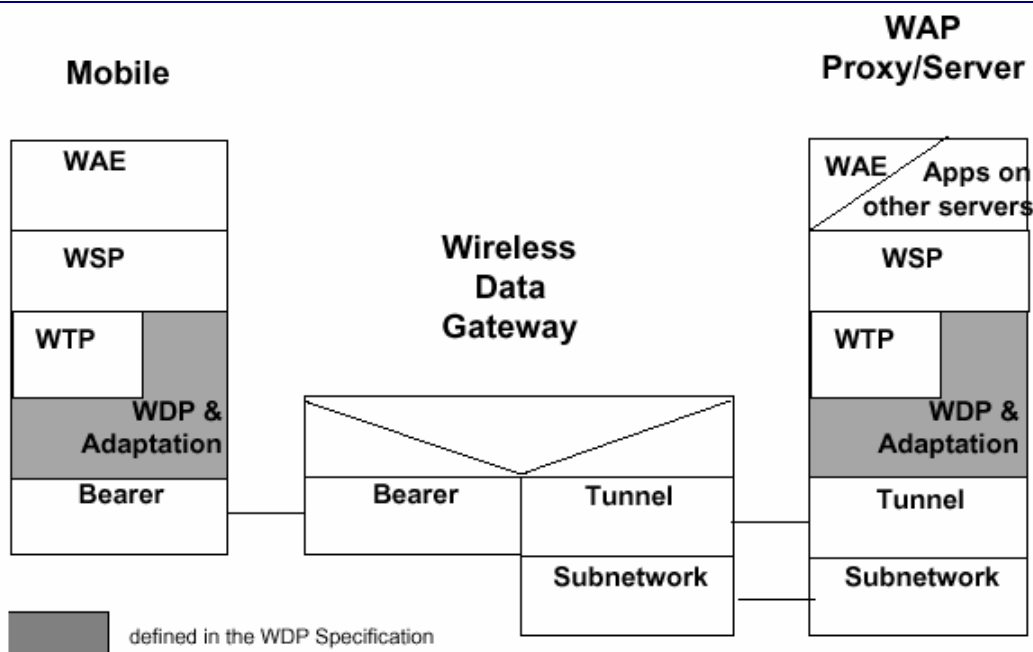
WDP-protokollen opererer over bærertjenestene i WAP. En av tjenestene WDP kan tilby er adressering ved bruk av portnummer, segmentering og reassemblering. I tillegg kan WDP om ønskelig lete etter feil i dataen som sendes.



Figur 7: Arkitekturen i WDP [10: WAP-forum, 1999]

WDP kan bruke flere bærere. Figur 7 illustrerer forskjellen i tjenestene som de forskjellige bærerne tilbyr og at WDP-protokollen må tilpasse sine tjenester til bærerne. Siden WDP-protokollen er optimalisert med tanke på blant annet båndbredde, kan ytelsen over de forskjellige bærerne variere.

Som en protokoll som støtter datagram tilbyr WDP sine tjenester til protokollene som ligger over i WAP-arkitekturen. WDP støtter mange kommunikasjonsprosesser fra høyere lag til bærertjenestene, selv om disse foregår samtidig. Portnummeret identifiserer det overliggende lagets komponent hvor dataen kommer fra. Det kan være en protokoll i seg selv eller det kan være en applikasjon som for eksempel elektronisk post.



Figur 8: WDP og dens relasjoner til WAP-arkitekturen [10: WAP-forum, 1999]

Tilpasningslaget (Adaption layer) er laget i WDP-protokollen som henviser WDP-protokollens funksjoner til en bestemt bærer. Dette laget fremstilles gjerne som et eget lag inne i WDP-protokollen. Dette illustreres i figur 8. Tilpasningslaget er forskjellig fra bærer til bærer. Når dataen kommer til en gateway blir den terminert og videresendt til WAP Proxy/Serveren. Det finnes også en tunnelprotokoll som er interface mellom gateway og WAP Proxy/Server.

3.5.1 Headerstruktur i WDP

Headerstrukturen for GSM SMS og GSM USSD i WDP er definert som UDH (User Header Data). GSM USSD støtter binære headere og GSM SMS støtter headere som er representert binært eller med tekst. Dette beskrives mer detaljert i kapittel 4.2.

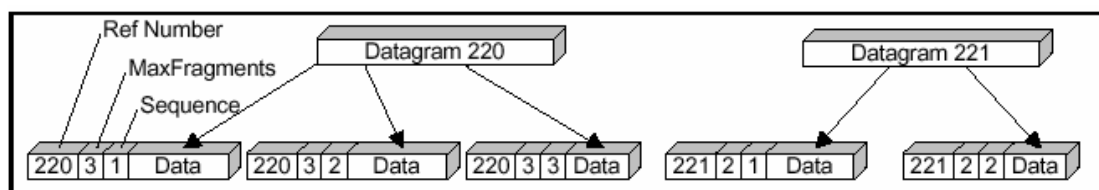
| FIELD | LENGTH |
|------------------------------------|-----------------|
| Length of User Data Header | 1 octet |
| Information Element Identifier 'A' | 1 octet |
| Length of Information-Element 'A' | 1 octet |
| Information-Element 'A' Data | 1 to 'n' octets |
| Information-Element-Identifier 'B' | 1 octet |
| Length of Information-Element 'B' | 1 octet |
| Information-Element 'B' Data | 1 to 'n' octets |
| ... | |
| Information-Element-Identifier 'n' | 1 octet |
| Length of Information-Element 'n' | 1 octet |
| Information-Element 'n' Data | 1 to 'n' octets |

Figur 9: Strukturen på UDH [10: WAP-forum, 1999]

Feltet med "Length-of-Information-Element" inneholder en integer som representerer antall oktetter som finnes "Information-Element-Data". Feltet "Length-of-User-Data-Header" inneholder en integer som representerer antall oktetter som finnes "User-Data-Header". Dette illustreres i figur 9.

3.5.2 Segmentering og reassemblering

Dersom størrelsen på pakkene overstiger kapasiteten på bærerene, må de deles opp i mindre deler. Dette illustreres i figur 10. Denne oppdelingen kalles for segmentering og gjøres før pakkene skal transporteres med den aktuelle bærerene. Når pakkene ankommer destinasjonen vil de bli satt sammen, slik at de blir sånn de opprinnelig var. Dette kalles reassemblering. I WDP finnes to forskjellige formater for segmentering, kort og lang form. Forskjellen på kort og lang form er størrelsen på "Datagram Reference Number" som brukes til å identifisere de segmenterte pakkene. Pakkene må identifiseres slik at de settes sammen i riktig rekkefølge ved destinasjonen.



Figur 10: Segmentering av datapakker [10: WAP-forum, 1999]

Headeren i en pakke inneholder:

- Referansenummer for WDP-pakken (0-255 eller 0-65535)
- Totalt antall segmenter i et datagram (maksimalt 255)
- Segmentnummer (1-255)

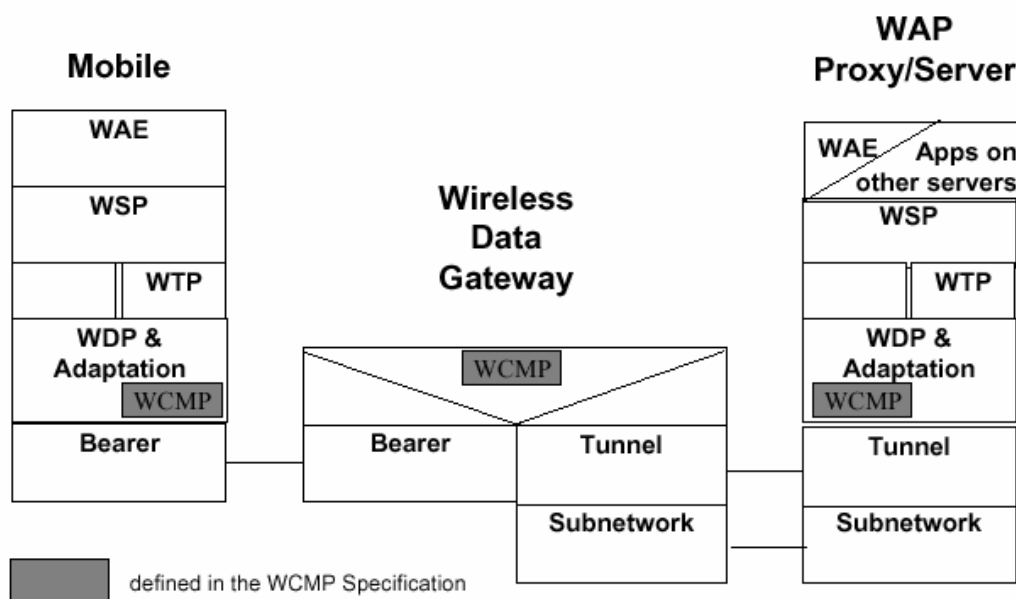
Maksimal lengde på et segmentert datagram av avhengig av pakkestørrelsen og bæreren. For GSM SMS er maksimal pakkestørrelse i nettverket 140 bytes og for GSM USSD er maksimal pakkestørrelse i nettverket 160 bytes [10: WAP-forum, 1999].

Sekvensnummeret kan brukes til å løse problemer med pakker som er dupliserte, mistet eller levert i feil rekkefølge. Sekvensnummeret er en teller som øker med en for hver pakke. Dersom en pakke skulle bli mistet, vil denne retransmitteres.

Reassembleringen foregår ved å bruke en liste med oversikt over pakkene som har ankommet. Når pakkene kommer settes de på riktig plass i listen. Listen blir deretter kontrollert slik at det blir bekreftet at pakkene er i riktig rekkefølge og at alle pakkene er der. Dersom listen stemmer, er datagrammet slik det opprinnelig var når det ble sendt, og det kan dermed transporteres til de overliggende lagene i protokollhierarkiet.

3.5.3 WCMP (Wireless Control Message Protocol) [22: WAP-forum, 1999]

WDP opererer over de forskjellige bærerne. I en overføring der WDP blir brukt kan det oppstå feil, og de øvre lagene i WAP kan ikke stole på WDP. For at kommunikasjonen i WAP skal fungere bedre bruker WDP WCMP. WCMP brukes til rapportere feil som kan oppstå under overføring av datagram.



Figur 11: WCMP i WAP-arkitekturen [22: WAP-forum, 1999]

WDP opererer over GSM SMS og GSM USSD. Feil som oppstår må rapporteres til WDP. Disse feilmeldingene må være unike og samtidig ende-til-ende kommunikasjon. Disse feilmeldingene er det WCMP som tar seg av. Feilene kan for eksempel være feil portnummer, for stort datagram, umulig å nå destinasjon, problemer med reassemblering eller parameterfeil i headeren til WDP.

4 WAP over GSM

4.1 GSM

Tjenestene i GSM deles inn i to hovedgrupper; teletjenester og bærertjenester. I tillegg til disse basistjenestene finnes tilleggstjenester som modifiserer eller supplerer en av basistjenestene. Eksempel på en slik tilleggstjeneste kan være viderekobling av en samtale.

Teletjenestene gir en komplett kommunikasjon, inkludert terminalfunksjoner mellom to brukere, i samsvar med standardiserte protokoller på alle nivåer. Bærertjenestene er overføring av brukerdata med gitte kapasiteter og egenskaper mellom definerte aksessgrensesnitt. Ren dataoverføring er et eksempel på en bærertjeneste, og det er opp til brukeren selv hvordan dataene skal brukes.

4.1.1 Telefoni

Brukerne har tilgang til nesten alle de samme fasiliteter som abonnentene i det faste telefonnettet.

Ved fullratekoding benyttes bittakten 13 kbit/s for taleoverføring. Dette er den høyeste netto bittakt som GSM-systemet tillater for taleoverføring. Fullrate dataoverføring har tilsvarende en netto bittakt på 12 kbit/s, og tillater inntil 9600 bit/s standard dataoverføring. Etter at signalet er kanalkodet, er brutto bittakt 22,8 kbit/s ved både fullrate tale- og dataoverføring [2: Bjugan, 1995].

Ved halvratekoding benyttes halve brutto bittakten, det vil si 11,4 kbit/s. To talekanaler/brukere tidsmultiplekseres til en fullratekanal og systemkapasiteten doubles. Halvrate tjenestene brukes ikke i Norge.

4.1.2 Kortmeldingstjenestene

Kortmeldingstjenestene (SMS – Short Message Service) gjør det mulig å sende meldinger med et begrenset informasjonsinnhold til eller fra mobilabonnenter i GSM. Det finnes tre typer kortmeldingstjenester:

- mobilterminert punkt til punkt
- mobiloriginert punkt til punkt
- kringkasting i radiocelle

En punkt-til-punkt melding kan ha en maksimal lengde på 160 7-biters tegn eller 140 oktetter. Punkt-til-punkt meldinger kan sendes eller mottas selv om mobilabonnenten

er opptatt i en samtale. En kringkastet kortmelding kan bestå av opptil 15 sider av inntil 93 tegn, og de kringkastede meldingene kan bare mottas når Mobilterminalen ikke er i trafikk [2: Bjugan, 1995].

Alle kortmeldinger må mellomlagres i et tjenestesenter SC (Service Center) som ligger utenfor GSM-nettet. Dette kan aksesseres på forskjellige måter, for eksempel via telefon, e-post eller via websider. Meldinger kan lagres her når abonnenten ikke er tilgjengelig (for eksempel utenfor dekningsområdet), og blir automatisk sendt når han/hun har radiokontakt igjen.

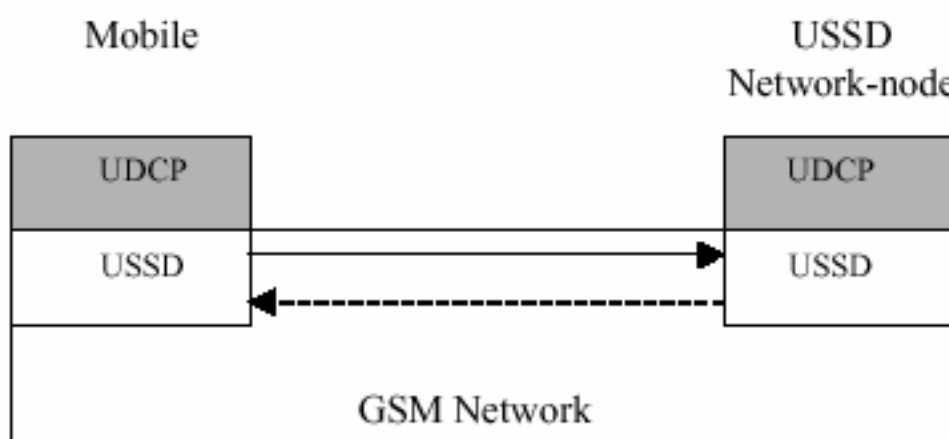
Punkt-til-punkt kortmeldinger kan for eksempel brukes

- når det ikke er mulig å få kontakt med en mobilabonnent i øyeblikket
- når en mobilabonnent er opptatt
- når det skal gis en beskjed på en ubeleilig tid av døgnet

4.2 UDCP (USSD Dialogue Control Protocol) [20: WAP-forum, 1999]

I GSM USSD (GSM Unstructured Supplementary Service Data) mangler det noen tjenester for støtte WAP-protokollene. GSM USSD er for eksempel bare halv dupleks og inneholder ingen destinasjonsadresse. For at GSM USSD skal støtte WAP brukes derfor tilleggstjenesten UDCP. UDCP-protokollen er i utgangspunktet ikke en del av GSM-spesifikasjonene fra ETSI (European Telecommunication Standardization Institute).

UDCP finnes i mobilterminalen mens USSD-nettverksnoden er i GSM-nettet. UDCP må uansett overlate adressen til en ekstern node. Dette kan for eksempel være en WAP gateway.



Figur 12: USSD og UDCP i GSM-nettverket

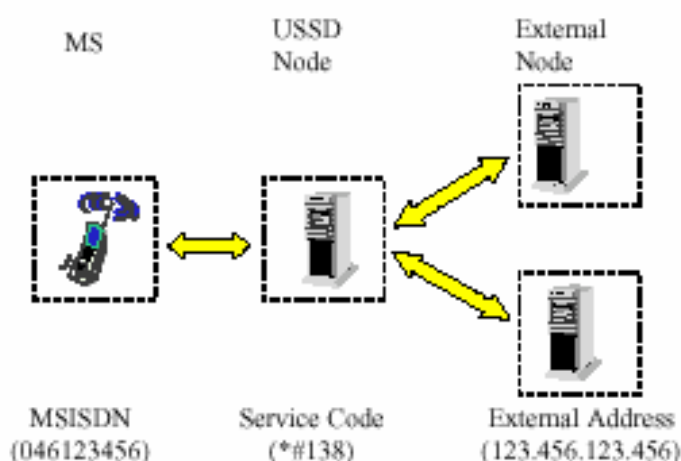
I mobilterminalen vil UDCP hele tiden lytte etter beskjeder fra USSD. Dette illustreres i figur 12. Det er vanlig at WDP bruker UDCP-protokollen.

4.3 Adresseringen i WAP [20: WAP-forum, 1999]

UDCP kan håndtere to typer nettverksarkitekturer og to adresseringsmetoder.

- A) Service Code i USSD Dialogue brukes til å adressere USSD-noden og den eksterne noden.
- B) Service Code i USSD Dialogue brukes til å adressere USSD-noden. En tilleggsadresse brukes i dette tilfellet til å adressere den eksterne noden.

B er i utgangspunktet mer aktuell enn A. Grunnen til dette er at dersom A skal sende data til en annen ekstern node må den eksisterende USSD-dialogen termineres og det må etableres en ny dialog. I B derimot er adressen til den eksterne noden inkludert i UDCP-headeren. Adressen til den eksterne noden blir sendt i hver operasjon som USSD utfører. Dette gjør det også mulig å være i dialog med flere eksterne noder samtidig.



Figur 13: USSD-noden adressert med Service Code og den eksterne noden adressert med en IP-adresse [20: WAP-forum, 1999]

Av figur 13 kan man se at den første operasjonen i USSD Dialogue kan inneholde data som skal til en ekstern node med IP-adressen 123.456.123.456. Den påfølgende operasjonen kan da inneholde data som skal til en annen destinasjon med en annen adresse. USSD-noden må også vite hvilken adressetype som skal brukes. Dette angis i en header når dialogen settes opp. Figur 14 viser en oversikt over de ulike adressetypene som kan brukes til denne adresseringen.

Overføring av prosessdata via WAP

| Address type | Assigned number |
|--------------|-----------------|
| IPv4 | 0x00 |
| IPv6 | 0x01 |
| GSM_MSISDN | 0x02 |
| Reserved | 0x03 to 0x07 |

Figur 14: Adresstypen som må angis til USSD-noden [20: WAP-forum, 1999]

5 WAP i fremtiden

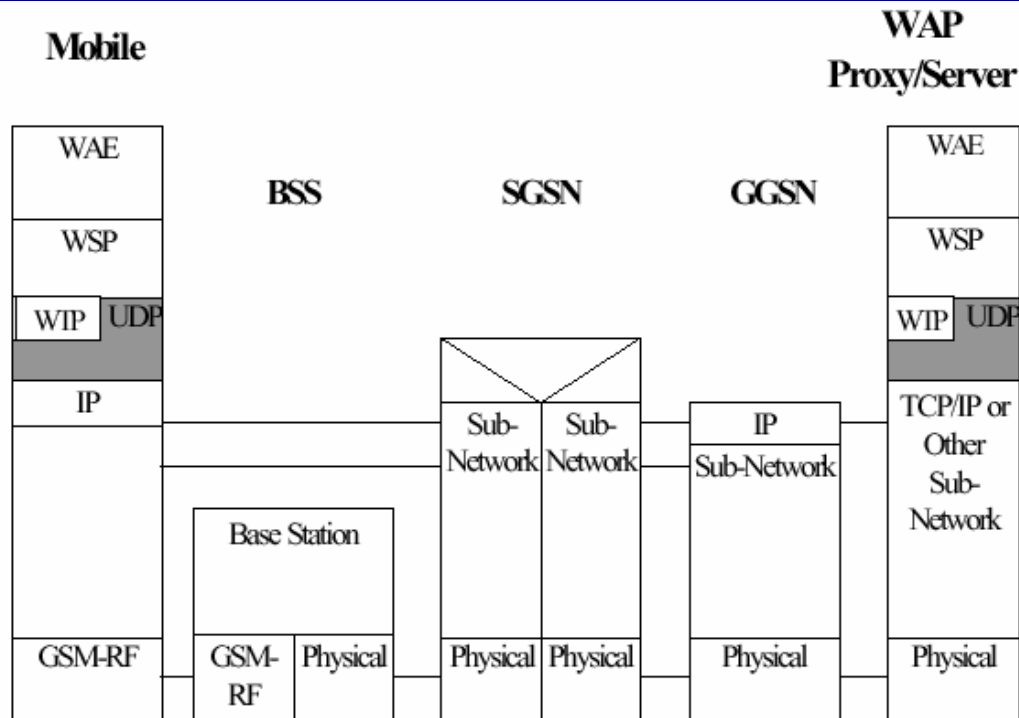
Om noen måneder vil raskere nettverk være tilgjengelige og nye muligheter åpner seg for WAP-teknologien. Den store utfordringen for WAP vil være på terminalsiden. For å kunne utnytte mulighetene som de nye nettverkene gir, må terminalene få større kapasitet blant annet når det gjelder minne og prosesseringskraft.

5.1 GPRS

GPRS (General Packet Radio Service) [3: Cai, Goodman, 1997, 4: Brasche, Walke, 1997] er en av de nye tjenestene som blir introdusert i GSM Fase 2+ standarden, og vil være tilgjengelig hos enkelte leverandører i slutten av 2000. Hensikten med GPRS er for å støtte periodevis og burst trafikk som man vanligvis har ved pakkeoverføring. GPRS deler og benytter seg av de samme GSM frekvensressursene og TDMA-strukturen (Time Division Multiple Access) som tale og linjesvitsjet datatrafikk bruker.

For å kunne oppnå høyere ytelse blir det i GPRS benyttet en metode som på en dynamisk og fleksibel måte deler de tilgjengelige ressursene med andre GSM tjenester. For å kunne overføre data ved høyere hastigheter blir det brukt flere tidsluker i stedet for en tidsluke per mobilstasjon som dagens GSM gjør. Antall tidsluker som blir benyttet blir regulert dynamisk ut i fra trafikkmengde, ledig kapasitet, og hva slags tjeneste brukeren ønsker og er villig til å betale for. Med GPRS kan man oppnå hastigheter opp mot 115kbit/s, i motsetning til dagens datatjeneste i GSM hvor man kun har linjesvitsjet dataoverføring med ytelse på 9,6 kbit/s.

For å tilføre denne funksjonaliteten i det eksisterende GSM nettet, blir det tilført noen nye noder i nettet for å støtte standard protokoller som IP og X.25. GPRS har støtte for både Punkt til Punkt (PTP) og Punkt til Multipunkt (PTM) tjenester, og tilbyr 4 nivåer av tjenestekvalitet (Quality of Service). GPRS er konstruert for å kunne reservere ressurser raskt, og ligger typisk på 0,5 til 1 sekund før noden kan begynne transmisjon av pakker.



Figur 15: WAP over GPRS [10: WAP-forum, 1999]

Som man kan se av figur 15 vil det komme inn noen nye noder når GPRS trer i kraft. GPRS støtter mobil IP-adressering og det er derfor UDP/IP som vil ta seg av de tjenestene som har med datagram å gjøre.

5.2 UMTS

UMTS (Universal Mobile Telecommunications System) [5: Grolms, 1999] er et tredje generasjons mobilkommunikasjons system som er under utvikling. Lanseringen av UMTS er beregnet til å skje i 2002.

Visjonen i UMTS er at den skal bli en fremtidig global plattform for mobilitet med en rekke nye tjenester og forbedringer i forhold til dagens 2. generasjons mobile systemer.

Disse områdene er:

- Forbedrede tjenester når det gjelder:
 - Talekvalitet, radiodekning, og kostnadseffektivitet
 - Tjenestekvalitet (QoS), transmisjonskvalitet, forsinkelse
 - Forbedret effektivitet, og kapasitet (Frekvensallokering og forbruk).

- Nye tjenester og funksjonaliteter:
 - Høy båndbredde for multimedia
 - Båndbredde etter behov, dynamisk allokering av båndbredde, asymmetrisk dataoverføring.
- Fleksibilitet med muligheter for å benytte flere aksessmetoder, frekvensområder, og teknologier.

5.3 De neste versjonene av WAP

I løpet sommeren år 2000 ventes det at mange av WAP-telefonene som kjøpes støtter versjon 1.2 av WAP. Denne versjonen vil inneholde noen nye tjenester som anses som viktig for utviklingen av WAP-teknologien. Blant tjenestene som vil komme i versjon 1.2 av WAP er WTA, Push, lokasjonskontroll og ende-til-ende-sikkerhet [23: Aslaksen, 2000].

WTA (Wireless Telephony Application) er en WAP-applikasjon som anvender telefonifunksjonalitet i nettet. Med WTA og WTAI er det mulig og konstruere viderekoblinger eller oppkoblinger som resultat av at brukeren har gjort bestemte valg. Et innkommende anrop kan gå via en WTA-server der oppringende part kan velge mellom tjenester som blir presentert på den oppringende persons WAP-telefon. En WTA kan også sende "events" når noe viktig har skjedd.

Push er en teknologi som gjør det mulig å sende ut meldinger dersom det skulle skje noe man ønsker å vite om. Eksempler på dette kan være tilbud på matvarer i en butikk som man om ønskelig får opp i displayet på WAP-telefonen når man går inn i butikken. Det kan være at aksjene man har steget eller sunket til et visst nivå. Hva man vil ha beskjer om styrer man selv ved å opprette brukerprofiler.

Når det gjelder lokasjonskontroll blir det mulig å se hvor en person med en WAP-telefon befinner seg. Dette blir gjort ved å registrere hvilken basestasjon som WAP-telefonen bruker. Størrelsen på området en basestasjon dekker er avhengig av befolkningstettheten. Dersom man for eksempel skulle bruke WAP-telefonen på en fotballkamp med mange personer innenfor et lite området, vil det være bort i mot umulig å finne personen som har brukt denne WAP-telefonen. Hvor god denne lokaliseringsfunksjonen er avhenger dermed på størrelsen på området som dekkes av basestasjonen.

Når det gjelder WAP versjon 2.0 kommer den sannsynligvis i begynnelsen av år 2001. Denne versjonen vil første rekke støtte nye tjenester for GPRS og 3G nett, og ikke være så relevant for eksempelvis vanlig GSM. Ellers vil sannsynligvis XHTML være markup språket [25: Kristensen, 2000].

I tillegg til at det arbeides med å lage nye spesifikasjoner for WAP-standarden, prøver Ericsson å lage en WAP-telefon med Bluetooth-interface. Ericsson har sammen med AU-system laget en demo-versjon som viser dette [27: WAP DEPLOYMENT FACT SHEET, 2000]

Det finnes ikke noe i de WAP-standardene som er under utvikling som støtter transparent overføring av data [25: Kristensen, 2000].

6 Beskrivelse av formater for overføring av data

6.1 WAV

WAV (Windows WAVE) er et filformat som er beregnet for lydfiler. Dette filformatet er utviklet av Microsoft og IBM [11: Boots-Ebenfield, Frumkes, 1998]. WAV er en del av Microsofts filformat RIFF som kan håndtere mange forskjellige typer data. RIFF er i utgangspunktet beregnet for multimediafiler som lyd- og bildefiler.

| Offset (hex) | Contents |
|-----------------|---|
| 0000 | 'R', 'I', 'F', 'F' |
| 0004 | Length of the entire file - 8 (32-bit unsigned integer) |
| 0008 | form type (4 characters) |
| 000B | first chunk type (4 character) |
| 0010 | first chunk length (32-bit unsigned integer) |
| 0014 | first chunk's data |
| ... | ... |

Figur 16: Struktur på filformatet RIFF [12: Weber, 2000]

En RIFF-fil (figur 16) er en stor bolk som er delt opp i mindre bolker som kan inneholde andre filtyper som for eksempel en WAV-fil.

| | | | |
|----|---------|----------------|---|
| 12 | 4 bytes | 'fmt ' | |
| 16 | 4 bytes | 0x00000010 | // Length of the fmt data (16 bytes) |
| 20 | 2 bytes | 0x0001 | // Format tag: 1 = PCM |
| 22 | 2 bytes | <channels> | // Channels: 1 = mono, 2 = stereo |
| 24 | 4 bytes | <sample rate> | // Samples per second: e.g., 44100 |
| 28 | 4 bytes | <bytes/second> | // sample rate * block align |
| 32 | 2 bytes | <block align> | // channels * bits/sample / 8 |
| 34 | 2 bytes | <bits/sample> | // 8 or 16 |

Figur 17: fmt beskriver samplingsformatet [12: Weber, 2000]

| | | | |
|----|---------|----------------------------|--|
| 36 | 4 bytes | 'data' | |
| 40 | 4 bytes | <length of the data block> | |
| 44 | bytes | <sample data> | |

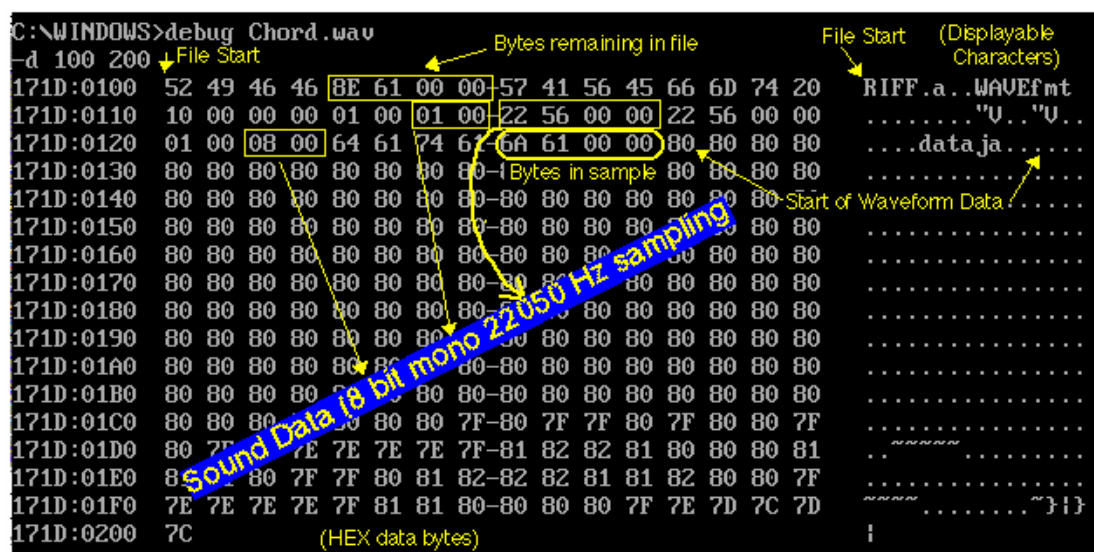
Figur 18: Datadelen i en WAV-fil [12: Weber, 2000]

En WAV-fil inneholder to bolker. Den ene bolken, fmt (figur 17), beskriver selve filen, mens den andre bolken (figur 18) inneholder samplene fra den lyden som har blitt digitalisert. Som man kan se av figur 17 inneholder filen blant annet samplingshastighet og lengde på bolken.

WAV støtter mange former for komprimering. Hvilken type komprimering som blir brukt blir angitt ved hjelp av en "Format tag". På figur 17 kan man se at "Format tag" har verdien 1. Dette vil si at komprimeringsalgoritmen som blir benyttet i dette tilfellet er pulskodemodulasjon (PCM) som er et av de vanligste formatene for å kode lyd.

Når det gjelder samplingsfrekvenser så er det mest vanlig å bruke 44.1 kHz (CD kvalitet), 22.05 kHz (radiokvalitet) og 11.025 kHz (telefonkvalitet) i en WAV-fil [13: McCarthy, 1998]. Andre frekvenser kan også benyttes. Ifølge "Nyquists samplingsteorem" er det vanlig å sample med en hastighet på minst det dobbelte av den høyeste registrerte frekvenskomponenten [14: Hammer, 1997]. Siden det hørbare frekvensområde er omtrent 20-20000 Hz for mennesker [15: Løvereide, 1998] samples lyd med CD kvalitet med en hastighet på 44.1 kHz.

Størrelsen på en WAV-fil er variabel og avhengig av innholdet. Uansett så er en WAV-fil i utgangspunktet stor. Det er ikke uvanlig med størrelser større enn 2.6 MB per minutt med lyd [16: PHS, 1999]. Størrelsen på filen avhenger også av om lyden skal være i mono eller stereo og antall sampler per sekund. Det er vanlig med 8 eller 16 sampler per sekund avhengig av hvor bra lydkort den aktuelle PC'en har. Med 16 sampler per sekund vil kvaliteten på gjengivelsen øke, men filen vil også bli større [13: McCarthy 1998].



Figur 19: Eksempel på WAV-fil [17: Gallant, 1997]

På figur 19 kan man se en WAV-fil i sin helhet. Det er i dette tilfellet en liten WAV-fil som er brukt som eksempel. I tillegg vil kvaliteten på denne filen være begrenset siden det er mono og 8 bit per sample.

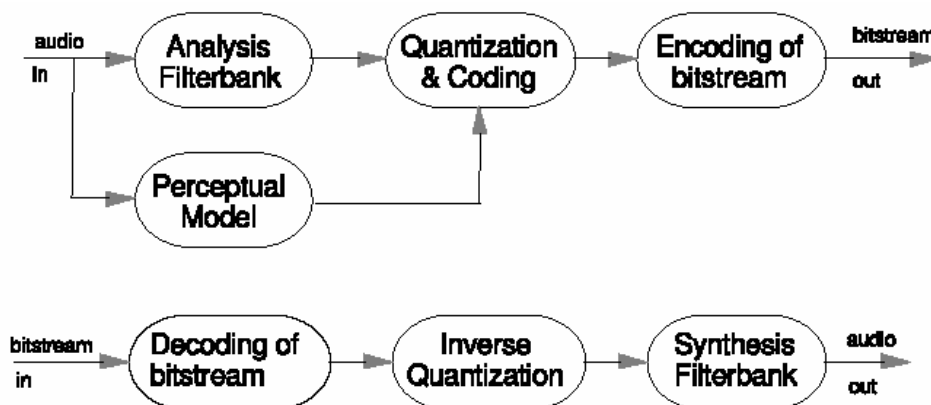
6.2 MP3

MP3 er et kodeformat som har hatt en rask utvikling siden starten i 1995. I 1999 var MP3 det mest brukte søkeordet på internett. MP3 er det mest brukte filformatet for lydfiler på internett. Denne filtypen er en av mange filformater utviklet av MPEG (Moving Pictures Experts Group) [The MPEG Home Page, mai 2000]. Alle kodeformater som er laget av MPEG baserer seg på forandring i lyd eller bilde. Det vil si at dersom den samme tonen spilles to ganger etter hverandre er det ikke noe forandring fra den første til den andre tonen, og kodingen kan dermed forenkles. Når det er avvik fra en tone til en annen, kodes dette ved å skrive inn forandringen i forhold til den første tonen til filen slik at man slipper å skrive all data om den nye tonen.

MP3 er egentlig MPEG-1/2 koding på lag 3. MP3 er en åpen standard. Det vil si at alle som er interessert kan implementere denne standarden. Verktøy for å kode og dekode MP3-filer har lenge vært lett tilgjengelig og kan lastes ned gratis fra internett.

Hensikten bak MP3 var å få et kodeformat der komprimeringen ble så effektiv som mulig. I tillegg så skal lyden etter kodingen være mest mulig lik lyden før kodingen. En viktig del av MP3 var at kodingen skulle være lite kompleks og så fleksibel som mulig. Grunnen til dette var for at det ikke skulle bli så kostbart å lage software og hardware som støttet kodingen.

Teknikken som brukes for å kode MP3 kalles "perceptual encoding". Dette er en teknikk som det har blitt forsket mye på siden slutten av 70-årene.



Figur 20: Blokkdiagram på et "perceptual encoding/decoding" system [21: Brandenburg, 1999]

MPEG audiokompresjon bruker mange forskjellige samplingsfrekvenser [21: Brandenburg, 1999]. MPEG-1 bruker frekvensene 32 kHz, 44.1 kHz og 48 kHz. MPEG-2 bruker halvparten av disse frekvensene, 16 kHz, 22.05 kHz og 24 kHz. MP3 sampler ved 8 kHz, 11.05 kHz og 12 kHz som er halvparten av samplingsfrekvensene i MPEG-2.

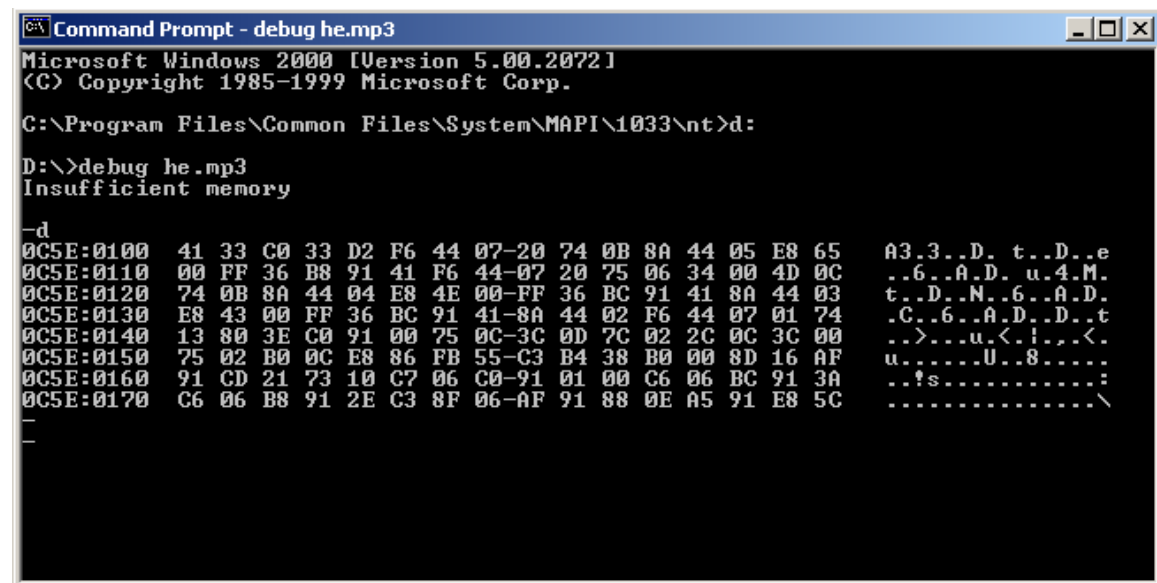
MPEG har ikke noe bestemt komprimeringsforhold. [21: Brandenburg, 1999]. Det finnes flere bitrater som kan brukes på den komprimerte lyden, avhengig av hvilken kvalitet man skal ha. MPEG-1 bruker 32 eller 16 kbit/s. MPEG-2 bruker 320 eller 160 kbit/s. I MP3 er det derimot, varierer bitraten fra ramme til ramme med lydsamplere. Denne variasjonen støttes av dekodeerne som skal tolke lydsignalene. Om det er ønskelig er det også i MP3-kodingen mulig å sette en fast bitrate.

MPEG-standarden har ingen garantier for kvaliteten på den komprimerte lyden [Brandenburg, 1999]. Kvaliteten på lydgjengivelsen baseres på blant annet bitraten til den komprimerte lyden og på de forskjellige koderne. Til forskjell fra vanlig Hi-Fi- og FM-utstyr vil koderne for MP3 (figur 20) lage feil og mangler i gjengivelsen dersom det blir brukt for liten bitrate i kodingen og/eller når det er feil på parameterne.

Feilene kan være:

- Forvrengning, men ikke harmoniske forvrengninger
- Støy, men bare ved bestemte frekvensområder
- ”hard lyd”, men feilen endrer karakteristikk hvert 20ms

En MP3-fil kan blant annet lages ved å komprimere en WAV-fil. Det finnes mye tilgjengelig utstyr som komprimerer en WAV-fil til MP3 på internett. Dette er ofte software som er gratis og kommer med varierende kvalitet. De fleste verktøyene som kan utføre en slik komprimering kan også foreta en reversibel funksjon, det vil si konvertere filen fra MP3 tilbake til WAV-formatet.



```

Command Prompt - debug he.mp3
Microsoft Windows 2000 [Version 5.00.20721]
(C) Copyright 1985-1999 Microsoft Corp.

C:\Program Files\Common Files\System\MAPI\1033\nt>d:

D:\>debug he.mp3
Insufficient memory

-d
0C5E:0100  41 33 C0 33 D2 F6 44 07-20 74 0B 8A 44 05 E8 65  A3.3..D. t..D..e
0C5E:0110  00 FF 36 B8 91 41 F6 44-07 20 75 06 34 00 4D 0C  ..6..A.D. u.4.M.
0C5E:0120  74 0B 8A 44 04 E8 4E 00-FF 36 BC 91 41 8A 44 03  t..D..N..6..A.D.
0C5E:0130  E8 43 00 FF 36 BC 91 41-8A 44 02 F6 44 07 01 74  .C..6..A.D..D..t
0C5E:0140  13 80 3E C0 91 00 75 0C-3C 0D 7C 02 2C 0C 3C 00  ..>...u.<.i...<.
0C5E:0150  75 02 B0 0C E8 86 FB 55-C3 B4 38 B0 00 8D 16 AF  u.....U..8.....
0C5E:0160  91 CD 21 73 10 C7 06 C0-91 01 00 C6 06 BC 91 3A  ..!s.....:
0C5E:0170  C6 06 B8 91 2E C3 8F 06-AF 91 88 0E A5 91 E8 5C  .....

```

Figur 21: Begynnelsen av en fil kodet i MP3

Figur 21 viser formatet til en MP3-fil. Dette er begynnelsen av filen som er en del av en sang kodet i MP3. Etter å ha komprimert 10 forskjellige WAV-filer til MP3-format ble det observert at MP3-filen ble omtrent 11 ganger mindre enn den opprinnelige WAV-filen i alle tilfeller (tabell 1). I en MP3 fil er det også mulig å legge til informasjon om sangen, det vil si artist og melodi. Denne informasjonen ligger som en header i begynnelsen av filen.

| | | | | | | | | | | |
|-------------|------|------|------|------|------|------|------|------|------|------|
| MP3 [MB] | 4,78 | 4,31 | 4,01 | 4,19 | 3,43 | 2,67 | 4,03 | 3,83 | 4,35 | 4,10 |
| WAV [MB] | 52,7 | 47,5 | 44,2 | 46,2 | 37,8 | 29,5 | 44,4 | 42,2 | 47,8 | 45,2 |

Tabell 1: Forhold i størrelsen mellom WAV- og MP3-filer

6.3 Sampler

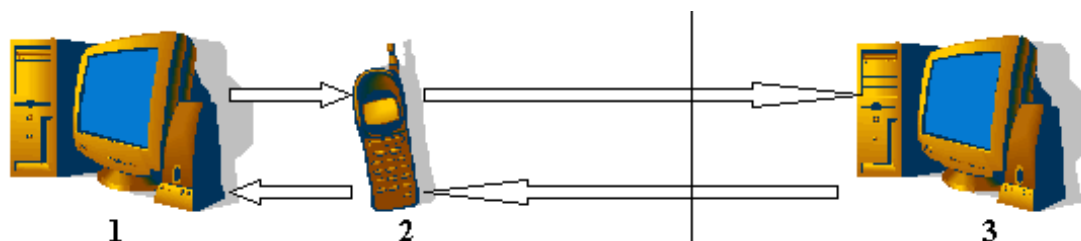
Et av alternativene for overføring er å overføre hvert sampel hver for seg. Hvert eneste sampel vil da beskrive et punkt på grafen (EKG-signalet) som skal overføres. EKG-signalet er i utgangspunktet et analogt signal.

Det analoge signalet blir digitalisert ved bruk av et A/D-kort som settes inn i en PC. A/D-kortet som brukes er PCM-DAS08. Dette kortet kan bruke en samplingsfrekvens på opp til 25 kHz. Dette 12 bits kort, det vil si at hvert sample kan presenteres på 2^{12} mulige nivåer. Samplingsfrekvensen som skal benyttes for å sample EKG-signalet er satt til 400 Hz. Dette er i tråd med "Nyquists samplingsteorem" som sier at det er vanlig å sample med en hastighet på minst det dobbelte av den høyeste registrerte frekvenskomponenten [14: Hammer, 1997]. Den høyeste registrerte frekvenskomponenten i et EKG-signal ligger på ca 150Hz [1: Jenkins, 1999].

Et tiltak som kan spare båndbredde er å pakke samplene i filer som reduseres ved hjelp av komprimering. Denne komprimeringen kan for eksempel være slik at den ser på antall like bits etter hverandre og komprimerer disse. Eksempel på kodeformater som kan brukes er Manchester-koding eller NRZI (Non-Return to Zero Inverted) [31: Peterson og Davie, 2000]. En slik koding vil gå ut over tidskravene siden det må ventes på et vist antall bits for å lage en fil av samplene. Filene bør være av en viss størrelse før de kodes, dersom kodingen skal bli effektiv. Det har for eksempel ikke noe særlig hensikt å kode 100 bits.

7 Metode

7.1 Systemskisse



Figur 22: Systemskisse over WAP-delen i oppgaven

Figur 22 viser en del systemet som skal utvikles. Systemet på figur 22 er bare en del av et større system som består av både bluetooth og WAP.

Utgangspunktet i oppgaven er en hjertepasient som selv skal kunne starte en alarm dersom han/hun føler seg dårlig. Dette skal kunne gjøres raskt og effektivt uten at pasienten må trykke på for mange knapper. Funksjonen som utløser alarmen må også kunne gjøres på en enkel måte siden mange er skeptiske og nøler med å benytte nye systemer.

Løsningen i sin helhet går ut på at pasienten som har WAP-telefonen også har Bluetooth-sender festet til kroppen. EKG-signalene blir målt ved at elektroder blir festet til pasientens kropp. Disse signalene vil først bli samlet og deretter overført fortløpende til en database (1) som befinner seg i en datamaskin med en Bluetooth-mottaker. Denne delen av løsningen blir behandlet i diplomoppgaven "Bluetooth – Overføring av EKG-prosessedata" [29: Grønhovd, 2000]

Dersom pasienten skulle føle seg uvel må han/hun utløse alarmen via WAP-telefonen (2). WAP telefonen skal da hente ut data fra databasen (1) og sende dataen videre til databasen i alarmsentralen (3). Det vil si at den ene databasen (1) befinner seg i nærheten av pasienten, mens den andre befinner seg i en alarmsentral på et sykehus. Sykehuset vil da kunne overvåke utviklingen til pasientens hjerterytme.

7.2 Teoretisk løsningsforslag

7.2.1 EKG-signal

Et EKG-signal er et signal med liten båndbredde. Båndbredden er på ca 0-150 Hz [Kap. 1.1.2]. Til sammenlikning er menneskets hørbare område på ca 20-20000 Hz

[Kap. 6.1]. Mye av lydutstyret som eksisterer er beregnet for dette frekvensområdet. Dette gjelder blant annet for originale CD-plater.

7.2.2 Databasene

Databasene skal bestå av to tabeller. Den ene tabellen kalles ”pasient” og skal bestå av informasjon om pasienten. Dette er informasjon som kan være nødvendig ved en eventuell alarm. Denne tabellen skal blant annet inneholde pasientens Etternavn, fornavn (og mellomnavn), adresse, postnummer, poststed, telefonnummer, primærlege og diagnose. I tillegg kommer Pasient_ID som er tabellens primærnøkkel. Pasient_ID skal være unik i tabellen. Primærnøkkel er et tall som settes automatisk ved å velge autonummer under Data Type. Informasjonen som ligger i den første tabellen skal er det første som skal sendes over til alarmdatabasen. Dette er viktig informasjon for de som har vakt ved alarmdatabasen. Dette gjelder særlig informasjon om hvor pasienten befinner seg.

I den andre tabellen kalles ”pasientfiler” og der vil det ligge filer/samplere som inneholder data fra pasientens EKG. For å holde orden på filene/samplene til en pasient er det en forbindelse mellom disse tabellene. Det er en one-to-many forbindelse. Dette vil si at en pasient kan ha mange filer/samplere registrert i sitt navn. Tabellene forbindes ved at Pasient_ID som er primærnøkkel i ”pasient” også finnes i ”pasientfiler” med tilsvarende egenskaper som i ”pasient”. Primærnøkkel i ”pasientfiler” kalles Fil_ID. Fil_ID settes da til autonummer. I tillegg til Pasient_ID og Fil_ID kommer Filnavn. Her skal navn settes på filene/samplene slik at de er lett å identifisere. Navnene skal gis i forhold til når andre filer/samplere som tilhører samme pasient ble lagret og hvilken filtype som er lagret. Et eksempel på et slikt filnavn kan være MP002, som angir at filtypen som er brukt er MP3 og at det er fil nummer 002 i en overføringssesjon som er lagret.

For å kunne aksessere databasene via WAP brukes ASP. ASP bygges inn i WML. ASP er et program beregnet for dynamiske websider. Inne i ASP-delen av programmet brukes SQL. SQL brukes blant annet for å hente ut og legge inn data i en database. Det er viktig at data i en database blir lagt og hentet fra riktig sted og dette er en av hovedfunksjonene til SQL som er et standard språk som fungerer for de fleste databaser.

Siden man hele tiden vet hva neste fil/sample som blir generert heter, kan det lages en funksjon i ASP som venter på at neste fil/sample skal bli lagret i databasen. Den riktige filen/sampelen vil da bli hentet ut med en gang en ny fil/sample er lagret. Dette kan gjøres ved å lage en løkke i ASP som hele tiden venter på neste fil/sample. Både filnavnet og en slik løkke kan lages ved å benytte en inkrementfunksjon, det vil si en funksjon der verdien av en variabel økes med 1 for hver løkke.

Databasen som befinner seg i pasientens hjem vil i utgangspunktet bare oppbevare data som har med pasienten selv å gjøre. Når pasienten utløser alarmen hjemme hos seg selv vil alle data om pasienten automatisk bli sendt til alarmdatabasen. Det er ikke noe behov for nærmere identifisering siden det bare er en pasient som bruker denne

databasen [Kap. 1.5]. Det som skjer når alarmen utløses på WAP-telefonen, er at WAP-telefonen tolker et WML-program som ligger på den samme datamaskinen som databasen. WAP-telefonen vil gå inn og lese programmet på en datamaskin som blir identifisert via en URL. En slik URL kan for eksempel bestå av IP-adressen til den aktuelle PC og hvor på PC'en programmet ligger.

I ASP skal det også lages en fil som setter opp forbindelsen til databasen. Denne filen kalles "conn.asp". Her må man deklarete hvilken driver som skal brukes og angi hvor databasene ligger.

7.2.3 Overføringsprosessen

Overføringen av data vil foregå slik at all informasjon om pasienten skal overføres først og deretter skal filene/samplene med pasientens EKG overføres. Informasjonen om pasienten ligger fast i databasen hjemme hos pasienten.

Når det gjelder overføringen av filer bør filene være små av størrelse, helst ikke mer enn 5 sekunder med EKG. Prosessdataen som skal overføres skal sendes med forholdsvis liten tidsforsinkelse, slik at man ikke har tid til å vente for lenge på at en fil skal produseres før den skal sendes. Filen bør heller ikke være for liten, siden den da vil inneholde en stor andel headere i forhold til prosessdata.

Databasene bør også fungere som buffere. Dersom programmet som skal lese EKG-signalet i alarmsentralen ikke har nye filer/samplere å lese som følge av en lengre tidsforsinkelse en beregnet, vil ikke det overførte signalet fungere som en helhet. Det er viktig at det ikke oppstår avbrudd i overføringen da hjerterytmen på et EKG-signal er meget viktig for de som evaluerer signalet. Fullrate dataoverføring har en netto bittakt på 12 kbit/s, og tillater inntil 9600 bit/s standard dataoverføring [Kap. 4.1.1]. Dette er en båndbredde som er for liten til å kunne overføre data i sanntid dersom man skal ta hensyn til sikkerhet i overføringene. Et av problemene med mobile nettverk er at tidsforsinkelsene er variable, samt at feltstyrken til senderne varierer så sjansen for avbrudd i overføringer er tilstede [Kap. 2].

Maksimal lengde på et segmentert datagram av avhengig av pakkestørrelsen og bæreren. For GSM SMS er maksimal pakkestørrelse i nettverket 140 bytes og for GSM USSD er maksimal pakkestørrelse i nettverket 160 bytes. Maksimum antall segmenter i et datagram er 255 [Kap. 3.5.2]. Dette gir en datagramstørrelse på maksimalt:

$140 \text{ bytes} * 255 \text{ segmenter} = 35700 \text{ bytes per datagram for GSM SMS}$

og

$160 \text{ bytes} * 255 \text{ segmenter} = 40800 \text{ bytes per datagram for GSM USSD}$

WAV-filer:

Observasjoner av EKG lagret som WAV-filer viser at et 5 sekunders opptak genererer en WAV-fil på mellom 430 og 480 kB. En WAV-fil på 480 kB gir

$$480 \text{ kB} * 1024 \text{ bits/kB} = 491520 \text{ bits}$$

GSM tillater en inntil 9600 bit/s standard dataoverføring. Dette gir

$$9600 \text{ bit/s} * 5 \text{ s} = 48000 \text{ bits som kan over i løpet av 5 sekunder.}$$

$$491520 \text{ bits} > 48000 \text{ bits}$$

Som man kan se av beregningene er det ikke mulig å overføre en WAV-fil som inneholder et EKG-signal via WAP. Overføringskapasiteten på GSM-nettet er for liten i forhold til datamengden som skal overføres med tanke på tidskravene en slik overføring krever.

MP3-Filer:

Man kan se at en WAV-fil på 480 kB er det samme som 491520 bits. Dersom en WAV-fil komprimeres til MP3-format vil filstørrelsen på MP3-fila statistisk sett være 10-12 ganger mindre enn den opprinnelige WAV-filen [Kap. 6.2].

$$491520 \text{ bits} / 10 = 49152 \text{ bits} > 48000 \text{ bits}$$

$$491520 \text{ bits} / 12 = 40960 \text{ bits} < 48000 \text{ bits}$$

Som man kan se av det første tilfellet så er også denne filen for stor for å kunne overføres dersom kravene til overføringstid skal kunne tilfredsstilles. Det andre tilfellet derimot vil kunne klare kravene.

Siden de fleste mennesker har forskjellig hjerterytme vil også filene som oftest bli av forskjellig størrelse, avhengig av hvem som testes. Dersom en person har rask hjerterytme vil det bli oftere forandringer i EKG-kurven. Siden MP3 kodes avhengig av forandringer, vil et slikt tilfelle gjøre komprimeringen fra WAV til MP3 dårligere. Det er uansett ikke store forskjeller, men dette må tas hensyn til siden dette er et grensetilfelle. I tillegg er det kun to personer som er med i testene der EKG-signalene ble tatt opp, og disse har ingen hjerteproblemer.

Som man kan se av resultatet er det mange MP3-filer som kan overføres slik at kravene tilfredsstilles. Når filene skal overføres vil det også bli lagt til noen headere som også må tas i betraktning når det gjelder overføringskapasitet. Headere er data som kommer i tillegg til fildataen. Et annet problem som har med båndbredde å gjøre er retransmisjon. Dersom et segment i et datagram skulle bli mistet, må dette segmentet overføres på nytt. Dette er en prosess som tar tid og som gjør at datagrammet må vente hos mottager før det kan sendes videre til de øvre lagene i WAP-hierarkiet [Kap. 3.5.2]. Dette er derfor et format som kan få problemer med å tilfredsstille de tidskravene som kreves ved overføring av EKG via en WAP-telefon.

Sampler:

A/D-kortet er stilt inn på en samplingsfrekvens på 400 Hz. Dette vil si at det samples 400 ganger hvert sekund. Hvert sample kan representeres med 12 bits.

Dette gjør at man trenger en overføringskapasitet på

$$400 \text{ Hz} \cdot 12 \text{ bits} = 4800 \text{ bits/s}$$

Dette er det overføringsformatet av de tre som har vært evaluert som krever minst båndbredde for å overføre data. Dette er også en størrelse som passer inn i et datagram for GSM SMS.

Valg av overføringsformat

Formatet som velges for overføring av prosessdata blir sampler. Dette er valget fordi sampler krever mindre båndbredde enn MP3 og WAV. I tillegg velges sampler fordi dette er den beste løsningen med tanke på en overføring med så liten forsinkelse som mulig.

Det vil også være mulig å få til en løsning med pakkesvitsjing. Når EKG-signalet sendes i form av sampler er det fortsatt mer plass til data i overføringskanalen. Denne plassen kan benyttes av andre datapakker som kan sendes i mellom datapakkene med sampler.

Siden sampler er valgt som overføringsformat, må det også lages et program som henter samplene fra databasen og setter de sammen til et EKG-signal. Dette kan være et helt enkelt program som formidler de digitale representasjonene på en synlig måte. Dette gjøres slik at hvert sampel er et målepunkt som vil bli representert med et punkt i et diagram. Dette diagrammet ville lage "rammen" for den synlige delen av programmet. I denne rammen skal punktene gradvis forskyves til venstre når nye punkter av etterfølgende sampler ble lest inn. Dette er en funksjon som kalles panning. Dette er en metode som er vanlig å benytte ved monitorering av signaler, i dette tilfellet et EKG-signal.

Den totale levetiden på et sampel i rammen kan bestemmes ved å sette parametere. Det er viktig at personellet i alarmsentralen får en god oversikt over pasientens EKG og levetiden på sampelet bør derfor være på mer enn 10 sekunder. Dette bør være tilstrekkelig for å få en oversikt over den siste tilstanden til en pasient.

Proessen foregår slik at man leser en fil eller sample fra databasen til en ramme. Da størrelsen på hvert sampel er på 12 bit, må hver ramme være av typen integer. Hver ramme måtte være så stort som man har punkter langs x-aksen i diagrammet. Til konstruksjonen av diagrammet kan putpixel (x, y, color)-funksjonen benyttes. For å få til en panning av diagrammet mot venstre trenger man bare å kopiere av et ramme-element til det foregående.

Det samme programmet vil det også kunne implementeres en funksjon som starter opp en lydfil som varsler alarm på alarmsentralen.

8 Valg av verktøy

Når man skulle velge verktøy var det mange valgmuligheter. Noe av utstyret ble valgt på forhånd, mens noe ble valgt underveis i prosjektet. Jeg har bevisst sett etter verktøy som støtter hverandre. Alt verktøy som er valgt støtter WAP, selv om noe av verktøyet krever mer båndbredde ved transmisjoner enn verktøy som er spesielt beregnet for WAP som for eksempel WML og WMLScript.

8.1 WAP utviklingsverktøy

Valget står i dette tilfellet mellom Ericsson WapIDE SDK 2.0 og Nokia WAP Toolkit v1.3 beta. Jeg velger Nokia siden jeg ikke får Ericssons utstyr til å virke. Grunnen til at Ericssons utstyr ikke fungerer er sannsynligvis fordi det ikke er kompatibelt med Windows 2000. De vil med ganske stor sikkerhet reagere forskjellig på enkelte koder og Nokias utviklingssider på internett er bedre enn Ericssons med tanke på denne oppgaven. Nokias sider omhandler mer av temaene som er til interesse for denne oppgave enn sidene til Ericsson.

8.1.1 WML

WML er et av språkene man kan bruke til å programmere WAP-applikasjoner. Her programmeres strukturen slik tjenestene skal presenteres for brukeren. Dette er et språk for å implementere brukergrensesnittet, håndtere innmating og spesifisere hva som skal skje ved ulike hendelser. Dette språket ble brukt for å gjøre det lett for pasienten å slå alarm. Språket ligner minner litt om HTML, men er det er mer kompakt. WBXML (WAP Binary XML) gjør at WML kan sendes effektivt over en WAP-forbindelse. WML sendes som en kompakt kode som oversettes av sender og mottaker i stedet for at det blir sendt tekststrenger.

8.2 Databasene

Databasene må kunne aksesseres med WAP-telefonen. Databasene som blir brukt er Microsoft Access 2000. Disse databasene bør kunne tilfredsstille alle kravene som er i denne oppgaven. Det vil si å kunne lagre data om pasienten, samt sampler til pasientens EKG.

8.2.1 ASP

For å kunne aksessere databasene ble ASP valgt. Grunnen til at dette ble valgt er at dette språket støtter WML. Jeg hadde også vært borte i språket tidligere og fant noen gode eksempler som jeg kunne ta i bruk. ASP er også språket som kan innføre logikk som for eksempel beregninger i oppgaven. Det må blant annet foretas en noen

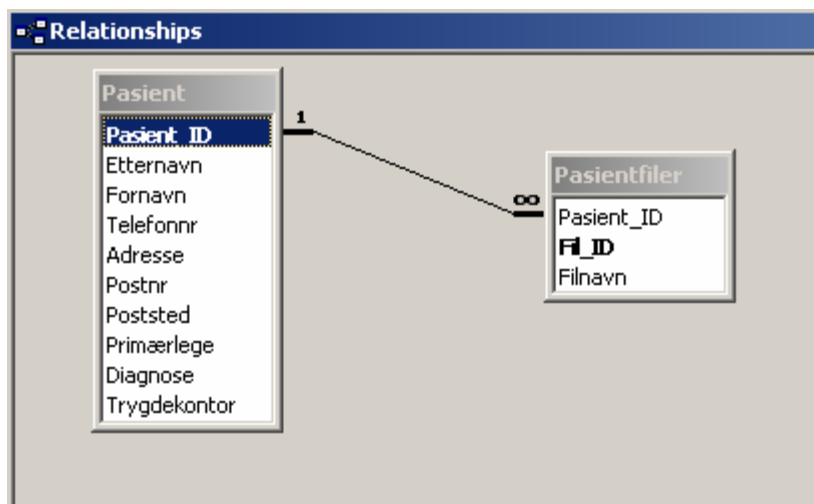
beregninger når samplene skal hentes ut av databasen slik at man får automatikk i prosessen og de riktige samplene blir hentet ut. Når man skal innføre logikk i et program beregnet for WAP kan man også bruke WMLScript.

8.2.2 SQL

Dette språket brukes for å hente data ut og sette data inn i databasen. Man må finne ut hvor i databasen samplene man skal ha er lagret og hvor man vil lagre de nye samplene man henter ut. Samplene vil bli lagret på samme sted i databasen, så det er ikke så mye SQL (Structured Query Language) som det er behov for.

9 Resultater

Det er laget to identiske databaser. Disse er lagt i forskjellige kataloger på PC'en. Den ene databasen skal forestille databasen i hjemmet til hjertepasienten. Den andre databasen er ment å være databasen på alarmsentralen.



Figur 23: Relasjonene i databasene

Som man kan se av figur 23 er det primærnøkkelen i "Pasient", Pasient_ID som knytter tabellene sammen. Man kan se av figuren at relasjonen er en one-to-many-relasjon. Hver pasient som er lagret i "pasient" kan da ha flere filer/sampler.

| Pasient : Table | | | | | | | | | | |
|-----------------|--------------|-----------|---------|-----------|---------|--------|----------|------------|---------------|--------------|
| | Pasient_ID | Etternavn | Fornavn | Telefonnr | Adresse | Postnr | Poststed | Primærlege | Diagnose | Trygdekortnr |
| + | 1 | Hansen | Harry | 99999999 | Heia 13 | 9999 | Hjemme | Dr. Olsen | Hjertekramper | Hjemme |
| ▶ + | 2 | | | | | | | | | |
| * | (AutoNumber) | | | 0 | | 0 | | | | |

Figur 24: Tabellen i databasen med data om pasientene

Tabellen i figur 24 inneholder data om pasienten som kan være viktig informasjon for en alarmsentral. Det er i denne tabellen plass til mange navn, men hjemmedatabasen skal i utgangspunktet bare inneholde informasjon om en person. Pasient_ID er primærnøkkel og settes som autonummer. Dette er et nummer som er unikt i tabellen og som vil settes automatisk ved at nummeret økes med en for hver pasient som legges inn i databasen.

| Pasientfiler : Table | | | |
|----------------------|------------|--------------|---------|
| | Pasient_ID | Fil_ID | Filnavn |
| | 1 | 1 | |
| ▶ | 0 | (AutoNumber) | |

Figur 25: Tabellen i databasen som inneholder pasientfilene

Tabellen i figur 25 inneholder alle filene til pasientene. I denne tabellen er Fil_ID primærnøkkel og satt som autonummer. Pasient_ID er et tall som brukes til å identifisere pasientene slik at filene sorteres til den pasienten som de kommer fra.

Når det gjelder brukergrensesnittet er dette designet slik at det skal være lett å forstå. Det er enkelt å sette i gang alarmen. Når alarmen er satt i gang finnes ingen funksjon for å avbryte den.



Figur 26: Displayet i WAP-telefonen forteller pasienten hva som må gjøres for å sette i gang alarmen

I displayet på WAP-telefonen står det "Velg Bekreft Alarm for å sette i gang alarmen". Pasienten må da trykke på knappen som tilsvarer valget "options" først. Når dette er gjort vil det komme frem et nytt bilde i displayet.



Figur 27: Displayet i telefonen ber pasienten bekrefte alarmen

I displayet på WAP-telefonen står det "Bekreft Alarm". Pasienten må da trykke på knappen som tilsvarer valget "Select". Når dette er gjort er alarmen satt i gang og det vil komme frem et nytt bilde i displayet som bekrefter dette (figur 28). Dersom pasienten ved et uhell skulle ha trykket feil kan han/hun gå tilbake til alarmmodus ved å trykke knappen som tilsvarer "Back". Displayet vil da gå tilbake til det opprinnelige skjermbilde som er vist på figur 26.



Figur 28: Bilde som bekrefter at alarmen er satt i gang

Under prosessen som blir utført av pasienten på WAP-telefonen vil det være en funksjon som henter ut data fra den ene databasen og setter dataen inn i den andre databasen. Det hele begynner med en asp-fil som setter opp forbindelsen, slik at dataene som skal hentes ut blir hentet fra riktig database. Denne filen er "Conn.asp" og ligger under vedlegg A

```
</card>

<card id="card1" title="ALARM">
  <p>
    <br/>
    <select name="Pasient">
      <option value="1">Hansen</option>
      <option value="2">Normann</option>
    </select>
  </p>
</card>
```

Figur 29: Koden som blir generert når prosessen for å hente ut av databasen kjøres

Figur 29 viser koden som blir generert av WAP-emulatoren når man forsøker å hente ut av databasen. Kildekoden som brukes i dette tilfellet ligger under vedlegg C og er kalt "Query.asp". Overskriften i WAP-telefonens display blir i dette tilfellet alarm. Når denne prosessen er fullført kan man velge mellom navnene Hansen og Normann som er hentet fra "Pasient" i databasen. Denne prosessen fungerer som den skal, selv om navnene ikke skal komme opp i displayet som en valgmulighet. Dette er gjort bare for å kontrollere at funksjonen som henter ut av databasen fungerer som den skal.

```
<card id="First_card">
  <do type="accept" label="Bekreft ALARM">
    <go href="#Second_Card"/>
  </do>

INSERT INTO Pasient (Fornavn,Etternavn) VALUES ('hermann','simpson')

  <p>
    Select <b> Next </b> to display the next card.
  </p>
</card>
```

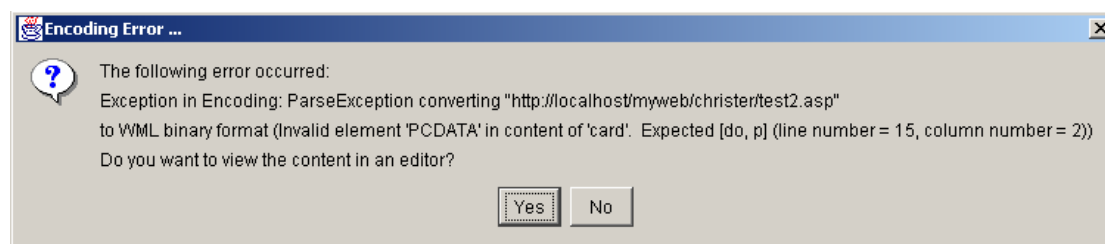
Figur 30: Kode som blir generert av WAP-emulatoren når det skal legges inn i databasen

På figur 30 kan man se koden genereres når man skal legge noe inn i databasen. Kildekoden som brukes i dette tilfellet ligger under vedlegg B og er kalt "Insert.asp". I dette tilfellet legges "hermann" inn i databasen under Fornavn og "simpson" legges inn i databasen under Etternavn. Det å hente ut data fra en database og legge inn data i en database ved bruk av WAP og WML fungerer i disse tilfellene som det skal.

Når man skal legge inn data i databasen får man likevel opp en feilmelding. Figur 31 viser hvordan denne feilmeldingen ser ut. Denne feilmeldingen kommer også opp når funksjonen som henter dataen ut av databasen og funksjonen som setter dataene inn i databasen kjøres i en og samme prosess.

Når data legges inn i en prosess blir riktig data lagt på riktig plass i databasen. Når man kjører begge prosessene samlet oppstår problemer. Dataen som blir hentet ut blir lagt i en variabel. Når data skal legges inn blir selve variabelfunksjonen lagt inn i databasen og ikke dataen som ligger i variabelfunksjonen.

Funksjonene som henter ut og legger inn data i databasen fungerer heller ikke feilfritt i sammen med funksjonene som setter i gang alarmen. I dette tilfellet kommer også den samme feilmeldingen som tidligere opp i skjermbildet.



Figur 31: Feilmelding ved forsøk på å samkjøre prosessene

Når det gjelder alarmsignal og program for å få samplene til EKG-signalet vist i en kurve så har ikke dette blitt gjort på grunn av tidsmessige årsaker.

10 Drøfting og diskusjon

10.1 Drøfting av resultater

10.1.1 Databasene

Databasene ble slik de var planlagt. En tabell inneholder informasjon om pasientene og en tabell inneholder samplene. Om databasene fungerer slik de er ment å gjøre er likevel usikkert siden det hverken er overført sampler eller filer. Det er uansett stor sannsynlighet for at databasene fungerer siden data fra tabellen som heter "Pasient" har blitt overført fra den ene til den andre databasen i form av to separate prosesser. Det er databasene som har skapt mest problemer. Problemene er i utgangspunktet små når man ser de i etterkant, men det er detaljer som det tar tid å finne ut av.

Når man setter opp en database har man i utgangspunktet kun leserettigheter dersom man bruker SQL. Man må altså justere rettighetene på databasene slik at man også får skriverettigheter når man bruker SQL. Siden det i utgangspunktet går an å lese fra databasene, men ikke skrive til de ved bruk av SQL, er det lett å tro at noe er feil med koden som har med SQLinsert å gjøre. Dette gjelder spesielt når feilmeldingen som kommer fram når applikasjonen startes ikke forklarer noe om hva som egentlig er galt. Får å få fram bedre og mer konstruktive feilmeldinger bør databasene aksesseres via internett først. Når man lager et webgrensesnitt får man mye bedre og mer detaljerte feilmeldinger. Det å aksessere databasene ved å bruke webgrensesnitt burde vært gjort tidligere enn det som var tilfelle i denne oppgaven. Dette kunne man spart mye tid på.

Et annet problem som dukket opp i forbindelse med databasene var primærnøklerne. Disse måtte være autonummererte, noe som på forhånd var vanskelig å vite.

10.1.2 Brukergrensesnittet

Når det gjelder brukergrensesnittet, så bør alle ord som står skrevet på skjermbildet være i et og samme språk. Prosedyrene for å sette i gang en alarm beskrives for brukeren på norsk, samtidig som beskrivelsen som gjør at brukeren kan navigere seg mellom de forskjellige skjermbildene står på engelsk.

I det første skjermbildet som kommer opp (figur 26) bør "Options" være byttet ut med "Alarm". Teksten bør da også forandres til "Velg "Alarm" for å sette i gang alarmen", slik at det er samsvar mellom tekst og valgmuligheter. På det andre skjermbildet som kommer opp (figur 27) etter at Options er valgt bør "Select" byttes ut med "Bekreft Alarm". Uansett så er funksjonen lett å utføre. Siden det bare er å trykke to ganger på samme knapp for å sette i gang alarmen er dette en prosedyre de fleste vil klare å utføre. Man kan derfor si at kompleksiteten ble så liten som det på forhånd var ønsket.

10.1.3 Overføringsformatet

Det å overføre sampler gjør at overføringen krever mye mindre båndbredde enn den ville gjort med MP3 eller WAV. Siden overføring av sampler bare benytter seg av omtrent 4800 bit/s, er det mulig å overføre annen data samtidig [Kap.7.2.3]. Man kan da benytte seg av multipleksing, noe som ville vært umulig dersom man måtte overføre MP3-filer. Det er også mulig å spare enda mer båndbredde ved å overføre sampler. Siden et EKG-signal ikke overstiger 150 Hz ville det vært mulig å sample signalet med en frekvens på 300 Hz [Kap. 6.3]. Det er også mulig å spare båndbredde ved å bruke et mindre antall bits til å representere hvert sample. Begge disse tiltakene og spesielt det siste, vil gå ut over kvaliteten på EKG-signalet som skal settes sammen i alarmsentralen. Et annet tiltak som kan spare båndbredde er å pakke samplene i filer som reduseres ved hjelp av for eksempel Manchester- eller NRZI-koding. Disse formene for komprimering fungerer slik at de ser på antall like bits etter hverandre og komprimerer disse [Kap. 6.3]. Et EKG-signal har en stor del av perioden en kurve som er flat. Dersom hvert punkt på dette flate partiet av kurven kan representeres med 12 nuller eller 12 1'ere, vil det være mulig å redusere dataen som skal overføres mye ved bruk av disse kodeformatene. Dette vil imidlertid gå ut over tidskravene siden det må ventes på et vist antall bits for å lage en fil av samplene [Kap. 6.3].

MP3 ville antakelig vært det beste alternativet dersom det hadde vært mulig å sette frekvensområdet som skulle kodes. MP3 er i utgangspunktet beregnet for menneskets hørbare frekvensområde fra 20-20000 Hz. Hadde man når man kodet signaler i MP3 kunnet satt opp parametere for 0-150 Hz kunne mye båndbredde vært spart. Det er likevel uvisst hvordan kvaliteten ville vært. Dersom kvaliteten hadde blitt like bra som det er i det hørbare frekvensområdet, ville MP3 med stor sannsynlighet vært det beste alternativet med sin gode komprimeringsevne.

Det å overføre WAV-filer via WAP med dagens båndbredde på overføringskanalene er helt utelukket.

10.2 Drøfting av verktøy

De verktøyene som er brukt i oppgaven fungerer bra i sammen. Et av problemene som oppstod var at man ikke kunne skrive ASP-kode i det samme vinduet som man kunne skrive WML. Løsningen på dette problemet ble derfor å bruke ASP Express og importere filene inn i WML-programmet ved å bruke "Load Location" på Nokias Toolkit. Å finne ut av dette var også noe det gikk med mye tid på. Dette skulle vise seg å bli en god løsning da man i Nokias Toolkit kunne lagre lokasjonen til de forskjellige filene som bokmerker.

10.3 Forbedringer

Det er en del forbedringer som kan gjøres siden det ikke har blitt overført prosessdata og det i tillegg er mye som ikke er gjort på grunn av dårlig tid. Uansett bør en av de mulige forbedringene være ved å lage et brukergrensesnitt som består av kun et språk.

På grunn av den dårlig båndbredden på mobiltelefoni kan man stille seg spørsmålet om det i hele tatt er noe grunn til å overføre prosessdata via en WAP-telefon. En bedre løsning ville vært å la alarmen bli satt i gang på WAP-telefonen, men overføre prosessdataen fra den ene databasen til den andre via fastnettet. Dette er en løsning som ville bedret overføringskapasiteten betraktelig, samtidig som overføringen blir sikrere. I et fastnett er det ingen avbrudd som følge av dårlig feltstyrke på signalet, samtidig som antall bitfeil i en dataoverføring er mye mindre for fastnett enn for nettverk beregnet for mobil kommunikasjon.

Slik som løsningen nå fremstår må brukeren av WAP-telefonen hele tiden være oppkoblet mot en bestemt lokasjon for å kunne starte alarmen. Det å konstant være oppkoblet krever mye strøm og batterikapasiteten er et svakt punkt på dagens mobile terminaler. Det bør derfor lages et brukergrensesnitt der pasienten ikke er konstant oppkoblet mot alarmfunksjonen, men der dette kan gjøres med få tastetrykk.

10.4 Drøfting av WAP i fremtiden

Siden WAP-teknologien er forholdsvis ny, vil det hele tiden foregå en utvikling der standarden for WAP utvides med nye funksjonaliteter.

Det er stor sannsynlighet for at det etter hvert kommer WAP-telefon med Bluetooth-interface [Kap. 5.3]. En slik WAP-telefon ville i dette tilfelle gjort at man hadde fått et system som hadde vært enklere å håndtere for en pasient. Man hadde dermed fått integrert alt utstyret pasienten trenger i et apparat.

Når det gjelder lokasjonskontroll, ville dette kunne føre til noen forbedringer i en alarmsituasjon. Nøyaktigheten på lokasjonen dårlig, og det ville ikke bli mye lettere å finne pasienten dersom han/hun for eksempel skulle være på besøk. En lokasjonskontroll kan likevel fortelle personellet på alarmdatabasen om pasienten er i nærheten av hjemmet sitt, og deretter gå ut i fra pasientens vanlige bevegelsesmønster dersom dette er kjent. Ved å se på roamingen mellom cellene som har blitt foretatt av basestasjonene kan man også se i hvilken retning pasienten beveger seg og med hvilken hastighet. Denne hastighetsmålingen vil ikke være så nøyaktig, men det vil for eksempel være lett å se om pasienten går eller bruker et kjøretøy. Dersom pasienten skulle befinne seg på et sykehjem, ville det sannsynligvis ikke vært nødvendig med lokasjonskontroll. Et sykehjem dekkes normalt med kun en basestasjon, siden det gjelder et område med lite areal.

Et punkt som ville gjort oppgaven mer fleksibel, er større båndbredde på overføringskanalen. GPRS eller UMTS er under utvikling og vil gi mobil

kommunikasjon mange nye muligheter. Båndbreddene disse nettene vil få, vil kunne gi brukerne av WAP en bedre grafisk presentasjon. Det ville i tillegg vært en fordel om det hadde eksistert en funksjon som støttet transparent overføring av data via WAP. Oppgaven ville sannsynligvis ikke fått noen andre resultater selv om disse forutsetningene hadde vært lagt til grunn for oppgaven. Grunnen til dette er at de nye funksjonalitetene som kommer i versjon 1.2 av WAP har med overføringsprosessen å gjøre. På grunn av tidsmessige begrensninger ble det ikke testet ut noen form for overføringsprosess, der båndbredden var den begrensende faktor.

11 Konklusjon

Oppgaven har bestått i å se på mulighetene for å overføre et EKG-signal via en WAP-telefon. I tillegg skulle forskjellige formater evalueres, for å se om noen av disse var mulig å bruke i en slik overføringsprosess.

Slik WAP-teknologien fremstår nå, oppnår man lite ved å overføre prosessdata via WAP. En av grunnene til dette er at WAP-standardene ikke støtter overføring av prosessdata via WAP.

Uansett hvilken versjon av WAP som brukes, ligger det for mange begrensninger i dagens GSM-nett. Man kan overføre et EKG-signal via GSM-nettet ved å bruke samplere. Et EKG-signal er et signal med liten båndbredde. Mye av prosessdataen som i praksis bør kunne overføres har en båndbredde som er mye større enn et EKG-signal. Dersom man skal overføre prosessdata med tre ganger så stor båndbredde som et EKG-signal vil man få problemer.

Konklusjonen blir at båndbredden på dagens GSM-nett er for liten til å overføre prosessdata via WAP. For at overføring av prosessdata via WAP skal bli vellykket må man ha bedre trådløse nettverk og funksjoner i WAP som støtter en slik overføring.

Når GPRS blir introdusert på markedet vil man få et trådløst nettverk som tåler de båndbreddene som kreves for overføring av prosessdata. Den beste løsningen for overføring av prosessdata vil på nåværende tidspunkt vil imidlertid være å overføre dataen ved å benytte et fastnett. Et fastnett tilbyr større båndbredder og overføring med færre antall bitfeil enn et GSM-nettet.

Referanser

- [1] Jenkins, Dean. **12-lead ECG homepage. The normal electrocardiogram.** 10. juli 1996, oppdatert 27. november 1999.
Webadresse: <http://homepages.enterprise.net/djenkins/ecghome.html> [10. februar 2000]
- [2] Bjugan, Vidar (1995). **GSM - Det globale systemet for mobilkommunikasjon (kap. 2)**. TAPIR, Universitetet i Trondheim.
- [3] Cai, Jian og Goodman, David J; Rutgers University (1997). **General Packet Radio Service in GSM (s 122-131)**, IEEE Communications Magazine.
- [4] Brasche, Götz og Walke, Bernhard; Aachen University of Technology (1997). **Concepts, Services and Protocols of the New GSM Phase 2+ General Packet Radio Service (s 94-104)**, IEEE Communications Magazine.
- [5] Grolms, Gabriela. **The role of Internet technology in UMTS**, Project I, Telenor R & D. 1999.
Webadresse: <http://infotorg.fou.telenor.no/prosjekter/umts/> [August 1999]
- [6] WAP-forum, **Wireless Application Environment Overview (s 10-22)**, 4. november 1999
Webadresse: <http://www.wapforum.org> [Februar 2000]
- [7] WAP-forum, **Wireless Session Protocol (s 11-14)**, 5. november 1999
Webadresse: <http://www.wapforum.org> [Februar 2000]
- [8] WAP-forum, **Wireless Transaction Protocol Specification (s 12-14)**, 11. juni 1999
Webadresse: <http://www.wapforum.org> [Februar 2000]
- [9] WAP-forum, **Wireless Transport Layer Security Protocol (s 12-15)**, 5. november 1999
Webadresse: <http://www.wapforum.org> [Februar 2000]
- [10] WAP-forum, **Wireless Datagram Protocol Specification (s 16-23, 42-45, 64-66)**, 5. november 1999
Webadresse: <http://www.wapforum.org> [Februar 2000]
- [11] Boots-Ebenfield, Marc og Frumkes, Lisa A. **Soundformats**, 19. november 1998, oppdatert 21. desember 1998.
Webadresse: <http://lang.swarthmore.edu/mellon/soundnvideo/soundformats.htm> [Mai 2000]
- [12] Weber, Timothy John. **The WAVE File Format**, 16. april 2000
Webadresse: <http://www.lightlink.com/tjweber/index.html> [Mai 2000]

- [13] McCarthy, Stephen. **Science – Computer Sound**, 8. februar 1998
Webadresse: http://www.sd68.bc.ca/mcge/div5/reports/computer_sound/report.htm
[Mai 2000]
- [14] Hammer, Øyvind. **Samlingsteoremet, aliasering**, 21. februar 1997
Webadresse: <http://www.notam.uio.no/arkiv/kurs/DIGLYD/node17.html> [Mai 2000]
- [15] Løvereide, Trond. **Introduksjon til lyd og musikk på datamaskin**, august 1998
Webadresse: <http://tor.hiof.no/~trondl/grit/lydogmusikk.htm#lydbehandling> [Mai 2000]
- [16] PHS Homepage. **C64S Frequently Asked Questions**, 97, 98, oppdatert 1. oktober 1999.
Webadresse: <http://www.phs-edv.de/c64s/doc/cfaqwav.htm> [Mai 2000]
- [17] Gallant M. **Wave Sound File Information**, 9. januar 1997
Webadresse: <http://204.191.136.6/~neutron/mmedia/index.html> [Mai 2000]
- [18] Paceart Associates, L.P. **EKG Speaks – Sample screens**, 22. oktober 1999
Webadresse: <http://www.ecgspeaks.com/index.html> [Mai 2000]
- [19] WAP-forum, **Wireless Application Protocol Architecture Specification (s 9-18)**, 30. april 1999
Webadresse: <http://www.wapforum.org> [Februar 2000]
- [20] WAP-forum, **WAP over GSM USSD Specification (s 9-19, 22-25)**, 15. juli 1999
Webadresse: <http://www.wapforum.org> [Februar 2000]
- [21] Brandenburg, Karlheinz. **MP3 and ACC explained**, Presentert 2-5. September 1999 AES 17th International Conference, Florence, Italia.
Webadresse: <http://drogo.cselt.stet.it/mpeg/tutorials.htm> [Mai 2000]
- [22] WAP-forum, **Wireless Control Message Protocol Specification (s 8-11)**, 4. august 1999
Webadresse: <http://www.wapforum.org> [Februar 2000]
- [23] Aslaksen, Erik. **Teknisk orientering fra WAP-forum; Hva vil fremtiden bringe?**, NORTIB WAP/SMS seminar mars 2000
Webadresse: <http://utvikler.mobilinfo.com/> [Mai 2000]
- [24] WAP-forum, **WAP White Paper**, oktober 1999
Webadresse: <http://www.wapforum.org> [Mai 2000]
- [25] Kristensen, Espen. Ericsson. Personlig meddelelse, 15 mai 2000.
Emailadresse: espen.kristensen@eto.ericsson.se

- [26] The MPEG Home Page, 3. juli 1996
Webadresse: <http://drogo.cselt.stet.it/mpeg/> [Mai 2000]
- [27] WAP DEPLOYMENT FACT SHEET, februar 2000
Webadresse:
http://www.motorola.se/wap/WAP_Deployment_Fact_Sheet_022000.doc [Mai 2000]
- [28] Telenor, ukjent publiseringsdato.
Webadresse: http://wap.telenor.no/go/docs/fase0/no/index.jsp?q='wh_wmlhtml' [Mai 2000]
- [29] Grønhovd, Per Rune. **Bluetooth – Overføring av EKG-prosessdata**, [publiseres juni 2000]. Diplomoppgave ved Høgskolen i Agder, avdeling for informasjons- og kommunikasjonsteknologi
- [30] Storfjord, Monica. **Security in WAP**, [publiseres juni 2000]. Diplomoppgave ved Høgskolen i Agder, avdeling for informasjons- og kommunikasjonsteknologi
- [31] Peterson, Larry L. og Davie, Bruce S. (2000). **Computer networks – A systems approach** (s 80-83), 2. utgave. Morgan Kaufman Publishers

VEDLEGG

Vedlegg A

Conn.asp

```
<%    option explicit

        Dim conn1    'connessione al database access
        Dim conn2

        Set conn1 = Server.CreateObject("ADODB.Connection")
        Set conn2 = Server.CreateObject("ADODB.Connection")

        'DSN-less connection to access
        ' conn.open "pasientinfo"
        conn1.Open "DRIVER={Microsoft Access Driver (*.mdb)};DBQ=" &
        Server.MapPath("hei/pasient.mdb") & ";"
        conn2.Open "DRIVER={Microsoft Access Driver (*.mdb)};DBQ=" &
        Server.MapPath("pasient.mdb") & ";"

%>
```

Vedlegg B

Insert.asp

```
<!--#include file="conn.asp" --><%
'send the right MIME type
Response.ContentType = "text/vnd.wap.wml"

Dim SQLquery, SQLinsert, SQLselect
Dim rsHei
%><?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
  <card id="First_card">
    <do type="accept" label="Bekreft ALARM">
      <go href="#Second_Card"/>
    </do>

  <%
SQLinsert="INSERT INTO Pasient (Fornavn,Etternavn) VALUES ('test','ljhg')"
conn1.Execute(SQLinsert)
response.write SQLinsert
%>

    <p>
      Select <b>Next </b> to display the next card.
    </p>
  </card>

  <card id="Second_Card">

    <p align="center">
      <b>
        <big>
          ALARM
        </big>
      </b>
    </p>
  </card>
</wml>
```

Vedlegg C

Query.asp

```
<!--#include file="conn.asp" --><%  
    'send the right MIME type  
    Response.ContentType = "text/vnd.wap.wml"  
  
    Dim SQLquery  
    Dim rsHei  
%><?xml version="1.0"?>  
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"  
    "http://www.wapforum.org/DTD/wml_1.1.xml">  
<wml>  
  
    <card id="splash" ontimer="#card1" title="Welcome to">  
        <timer value="50"/>  
  
    </card>  
  
    <card id="card1" title="choose a film">  
  
    <%  
        SQLquery = "SELECT [Pasient_ID], [Etternavn] FROM Pasient"  
        set rsHei = conn.Execute(SQLquery)  
  
        if rsHei.eof then  
            rsHei.close  
            set rsHei = nothing  
            Response.write("Sorry, no Hei today")  
            Response.write("</p></card></wml>")  
            Response.end  
        end if  
  
    %>  
  
    <p>  
        <br />  
    <select name='Pasient'>  
    <%  
        Do while not rsHei.eof  
            response.write("<option value="" & rsHei("Pasient_ID") & "">" &  
rsHei("Etternavn") & "</option>" & vbcrLf)  
            rsHei.MoveNext
```

```
    loop %>
</select>
```

```
</p>
<% rsHei.close
    set rsHei = nothing
%>
</card>
</wml>
```

Vedlegg D

Forkortelser

ASP - Active Server Pages
EKG - Electrocardiogram
FM- Frekvens Modulasjon
GPRS - General Packet Radio Service
GSM - Globale System for Mobile Communication
Hi-Fi - High Fidelity
HTML - Hypertext Markup Language
MP3 - MPEG Layer 3
MPEG – Moving Pictures Experts Group
NRZI - Non-Return to Zero Inverted
PDA - Personal Digital Assistant
RAM - Random Access Memory
ROM - Read Only Memory
SMS - Short Message Service
SQL - Structured Query Language
UDCP - USSD Dialogue Control Protocol
UDH - User Header Data
UMTS - Universal Mobile Telecommunications System
URL - Uniform Resource Locator
USSD - Unstructured Supplementary Service Data
WAP - Wireless Application Protocol
WAV – Betegnelse på lydfilformat for Windows.
WBMP - Wireless Bitmap
WBXML - WAP Binary XML
WML - Wireless Markup Language
WTA - Wireless Telephony Application
WWW - World Wide Web
XML - Extensible Markup Language
XHTML – Extended HTML